



**Escuela Superior  
de Ingeniería y Tecnología**  
Universidad de La Laguna

# Trabajo de Fin de Grado

Grado de Ingeniería Informática

---

## Tecnologías Fiware aplicadas a la gestión inteligente de residuos

*Fiware technologies applied to smart waste management*

Cristian González Alonso

---

La Laguna, 8 de Julio de 2022

D. **Julio Antonio Brito Santana**, con N.I.F. **42812193Q** profesor Titular de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor

## **C E R T I F I C A**

Que la presente memoria titulada:

***“Tecnologías Fiware aplicadas a la gestión inteligente de residuos”***

ha sido realizada bajo su dirección por D. **Cristian González Alonso**  
con N.I.F. **45866856A**.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en, La Laguna a 8 de julio de 2022.

# Agradecimientos

Agradezco a **mis padres y mi hermano**, por haber estado siempre conmigo durante todo este proceso que, a pesar de haber sido más largo de lo esperado, ha sido muy enriquecedor y del cuál me llevo muy buenos recuerdos.

Agradezco también a mi pareja **Itziar**, por su apoyo incondicional durante todo el duro y largo proceso de elaboración de este proyecto, y a mi gato **Barraquito**, por todas esas tardes haciéndome compañía mientras redactaba esta memoria.

Gracias a mis compañeros de carrera y amigos, **Iván, Pablo, Dani e Iru**, que a pesar de que ellos acabaron antes que yo, siempre estuvieron cuando los necesitaba y a día de hoy lo siguen estando.

Por último, pero no menos importante, agradecimientos a mi tutor **Julio Brito Santana**, por su apoyo durante todo el periodo que ha abarcado este TFG y por confiar en mí y querer continuar a mi lado a pesar de haber empezado este proyecto el año pasado.

# Licencia

\* Si quiere permitir que se compartan las adaptaciones de tu obra mientras se comparta de la misma manera y NO quieres permitir usos comerciales de tu obra indica:



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional.

# Resumen

La gestión eficiente de residuos y en particular los servicios urbanos de recogida son retos importantes a los que se enfrentan las ciudades. La gestión de residuos es un pilar fundamental de la economía circular, de las ciudades y destinos turísticos inteligentes. La gestión de residuos urbanos la componen distintas etapas, que van desde la recolección y el transporte hasta su tratamiento. La recolección de restos sólidos es una actividad clave, del sistema de gestión, en la mejora del servicio al ciudadano, en la reducción de impactos medioambientales y en el ahorro de costes en la administración.

En base a estos retos, este Trabajo Final de Grado desarrolla una plataforma de captura, tratamiento eficiente y visualización de datos de la gestión de residuos, para facilitar la gestión del conocimiento y la toma de decisiones. Esta plataforma está basada en la integración de tecnologías Fiiware con otras aplicaciones que mejoran sus prestaciones.

En un futuro, la información disponible y la plataforma podrá utilizarse para la implementación de soluciones inteligentes. Soluciones a los problemas de organización y planificación existentes en las ciudades, asociados a la gestión de los residuos que se generan en ellas.

**Palabras clave:** Fiiware, gestión de residuos, Qgis, PowerBi, smart cities, Inteligencia de Negocios, sistemas de información, datos.

## Abstract

Improving the urban collection service and more efficient waste management are major challenges that cities face nowadays. Waste management is a fundamental pillar of the circular economy, of smart cities and tourist destinations. It is made up of different stages, from collection and transport to treatment. The collection of solid waste is a key activity of the management system, in the improvement of the service to the citizens, the reduction of environmental impacts and in the saving of administration costs.

Based on these challenges, this Final Degree Project develops a platform for the capture, efficient treatment and visualization of waste management data, to facilitate knowledge management and decision-making. This platform is based on the integration of Fiware technologies with other applications that improve its performance.

In the future, the information available and the platform itself can be used for the implementation of intelligent solutions. Solutions to existing organization and planning problems in cities, associated with the management of waste generated in them.

**Keywords:** Fiware, waste management, Qgis, PowerBi, smart cities, Inteligencia de Negocios, information system, data.

# Índice general

<b>Capítulo 1: Introducción</b>	<b>1</b>
1.1 Antecedentes	1
1.2 Objetivos	2
1.3 Desarrollo y tecnologías	3
1.4 Organización de la memoria	4
<b>Capítulo 2: Fundamentos</b>	<b>5</b>
2.1 La importancia de los datos	5
2.2 Inteligencia de Negocios	6
2.3 Sistemas de información	7
2.4 SmartCities	8
2.5 Gestión de residuos	9
<b>Capítulo 3: Tecnologías Fiware</b>	<b>11</b>
3.1 Plataforma Fiware	11
3.3 Arquitectura y componentes	12
3.4 Modelos de datos	15
3.5 Casos de uso	16
<b>Capítulo 4: Diseño y prototipado de una plataforma orientada a datos para la gestión de residuos.</b>	<b>18</b>
4.1 Arquitectura	18
4.2 Integración de componentes tecnológicas	21
4.2.1 Google Cloud Platform	21
4.2.2 Scripts de extracción	22
4.2.3 Modelo de datos y habilitadores Fiware utilizados	26
4.2.4 Qgis	29
4.2.5 PowerBi	37
4.2.6 Wordpress	40
<b>Capítulo 5: Conclusiones y líneas futuras</b>	<b>43</b>
<b>Capítulo 6: Summary and Conclusions</b>	<b>45</b>
<b>Capítulo 7: Presupuesto</b>	<b>46</b>
7.1 Software como servicio	46
7.2 Coste del personal	47

7.3 Presupuesto total	47
<b>Capítulo 8: Algoritmos</b>	<b>48</b>
8.1 Algoritmo : Extracción de datos abiertos	48
8.2 Algoritmo : Procesado y carga de los datos	50
<b>Bibliografía</b>	<b>57</b>



# Índice de figuras

<a href="#"><u>Figura 1.1:</u></a>	Etapas de trabajo	2
<a href="#"><u>Figura 3.1:</u></a>	Logo Fiware	11
<a href="#"><u>Figura 3.2:</u></a>	Empresas miembro de la comunidad Fiware	12
<a href="#"><u>Figura 3.3:</u></a>	Servicios Fiware levantados en Docker	13
<a href="#"><u>Figura 3.4:</u></a>	Arquitectura y funcionamiento del Context Broker	13
<a href="#"><u>Figura 3.5:</u></a>	Arquitectura de la plataforma Fiware	14
<a href="#"><u>Figura 3.6:</u></a>	Visualización de datos de Badajoz es más	17
<a href="#"><u>Figura 4.1:</u></a>	Diseño de la arquitectura de la plataforma	19
<a href="#"><u>Figura 4.2:</u></a>	Logo Google Cloud Platform	21
<a href="#"><u>Figura 4.3:</u></a>	Logo de Python	22
<a href="#"><u>Figura 4.4:</u></a>	Script de extracción de datos en crudo (I)	23
<a href="#"><u>Figura 4.5:</u></a>	Script de extracción de datos en crudo (II)	23
<a href="#"><u>Figura 4.6:</u></a>	Script de procesado de datos crudos(I)	24
<a href="#"><u>Figura 4.7:</u></a>	Script de procesado de datos crudos(II)	24
<a href="#"><u>Figura 4.8:</u></a>	Datos en BigQuery	25
<a href="#"><u>Figura 4.9:</u></a>	Estructura de la entidad WasteContainer	27
<a href="#"><u>Figura 4.10:</u></a>	Estructura de la entidad Device	28
<a href="#"><u>Figura 4.11:</u></a>	Funcionamiento habilitador Cygnus	29
<a href="#"><u>Figura 4.12:</u></a>	Logo QGIS	29
<a href="#"><u>Figura 4.13:</u></a>	Mapeado de población con QGIS	30
<a href="#"><u>Figura 4.14:</u></a>	Selector de tema en el plugin del ISTAC en QGIS	30
<a href="#"><u>Figura 4.15:</u></a>	Selector de indicador en el plugin del ISTAC en QGIS(I)	31
<a href="#"><u>Figura 4.16:</u></a>	Selector de indicador en el plugin del ISTAC en QGIS(II)	31
<a href="#"><u>Figura 4.17:</u></a>	Mapa de Gestión de Residuos de Badajoz	32
<a href="#"><u>Figura 4.18:</u></a>	Mapeado extensión de Adeje	33
<a href="#"><u>Figura 4.19:</u></a>	Mapa de Adeje	33
<a href="#"><u>Figura 4.20:</u></a>	Mapeado El Galeón con ubicación de cubos	34
<a href="#"><u>Figura 4.21:</u></a>	Mapeado con ubicación contenedores amarillos	35
<a href="#"><u>Figura 4.22:</u></a>	Mapeado con ubicación contenedores verdes	35
<a href="#"><u>Figura 4.23:</u></a>	Logo de PowerBi	38
<a href="#"><u>Figura 4.24:</u></a>	Cuadro de mando sobre contenedores y retirada de vidrio realizado en PowerBi	39

<b><u>Figura 4.25:</u></b> Cuadro de mando de frecuencia de reciclaje en habitantes	39
<b><u>Figura 4.26:</u></b> Logo Wordpress	40
<b><u>Figura 4.27:</u></b> Mockup página web (I)	41
<b><u>Figura 4.28:</u></b> Mockup página web (II)	41
<b><u>Figura 4.29:</u></b> Mockup página web (II)	42

# Índice de tablas

<a href="#"><u>Tabla 3.1:</u></a>	Habilitadores opcionales de Fiware	14
<a href="#"><u>Tabla 3.2:</u></a>	Dominios de datos de Fiware	15
<a href="#"><u>Tabla 4.1:</u></a>	Servicios utilizados de Google Cloud Platform	21
<a href="#"><u>Tabla 4.2:</u></a>	Entidades de la entidad WasteManagement	26
<a href="#"><u>Tabla 4.3:</u></a>	Etiquetas de la entidad WasteContainer	26
<a href="#"><u>Tabla 4.4:</u></a>	Iconografía QGIS	36
<a href="#"><u>Tabla 7.1:</u></a>	Presupuesto tecnologías	45
<a href="#"><u>Tabla 7.2:</u></a>	Presupuesto por coste del personal	46
<a href="#"><u>Tabla 7.3:</u></a>	Presupuesto total	46

# Capítulo 1 : Introducción

## 1.1 Antecedentes

La **Inteligencia Empresarial** o de Negocios (**Business Intelligence**) es el conjunto de técnicas y conocimientos empleados para transformar la información en conocimiento, el cual se utilizará posteriormente en la toma de decisiones de una organización. El propósito es obtener conocimiento para el establecimiento de medidas en la organización conducentes a la reducción de costes y tiempos en el desarrollo de las actividades (beneficios tangibles), o la mejora de la posición competitiva y un mayor conocimiento disponible en la organización (beneficios intangibles).

En general la información que poseen las organizaciones está gestionada por los denominados **Sistemas de Información (SI)**. Estos son conjuntos de datos, procesos, personas que interactúan entre sí ayudando a su almacenamiento, tratamiento y distribución de manera que sean útiles para la organización. Todas las organizaciones disponen de sistemas de información con más o menos automatización, integración y complejidad. Ahora bien, lo importante es el uso eficiente de la información disponible para hacer predicciones, mejorar la toma de decisiones y tomar ventajas competitivas aprovechando las capacidades tecnológicas disponibles en la actualidad. Actualmente los SI en general incorporan o están asociados al Inteligencia de Negocios, pues los primeros sirven de soporte para la toma de decisiones que se lleva a cabo en el segundo.

En la actualidad, la cantidad de datos que se generan dentro y fuera de las organizaciones, y están disponibles, aumentan de forma exponencial y con la implantación de tecnologías emergentes (IoT, Móviles, Redes Sociales, 5G,...) aún más. Las herramientas tecnológicas disponibles para el almacenamiento y el procesamiento (Big Data, Cloud,...) de estos datos están al alcance de la mayoría de organizaciones públicas y privadas facilitando nuevas soluciones, productos, servicios, negocios y modelos de valor ( Industria 4.0, Smart Cities,..). En este contexto la innovación y la transformación digital es imparable para todas las organizaciones, ciudades y territorios. El acceso, generación, disponibilidad, divulgación, uso y explotación de los datos e información es una estrategia clave para el desarrollo competitivo.

Fiware es una plataforma, impulsada por la Unión Europea, para el desarrollo y despliegue de aplicaciones en Ciudades Inteligentes, con el objetivo de crear soluciones inteligentes a los diversos problemas que se puedan presentar. Estas soluciones tienen como particularidad que recogen información del entorno en el que se desenvuelven desde distintas fuentes, tratando de obtener una respuesta lo más eficiente y adecuada posible para la resolución del problema.

Este Trabajo de Fin de Grado propone el diseño y prototipado de una plataforma que permita el tratamiento de datos relacionados con la gestión de residuos. El propósito de esta plataforma es la captura y disposición de datos otorgando la posibilidad al usuario la visualización de los mismos de manera gráfica y sencilla. Además, que permita integrar de forma modular en el futuro, procedimientos inteligentes de planificación, optimización y toma de decisiones que hagan uso de los conocimientos disponibles en la plataforma.

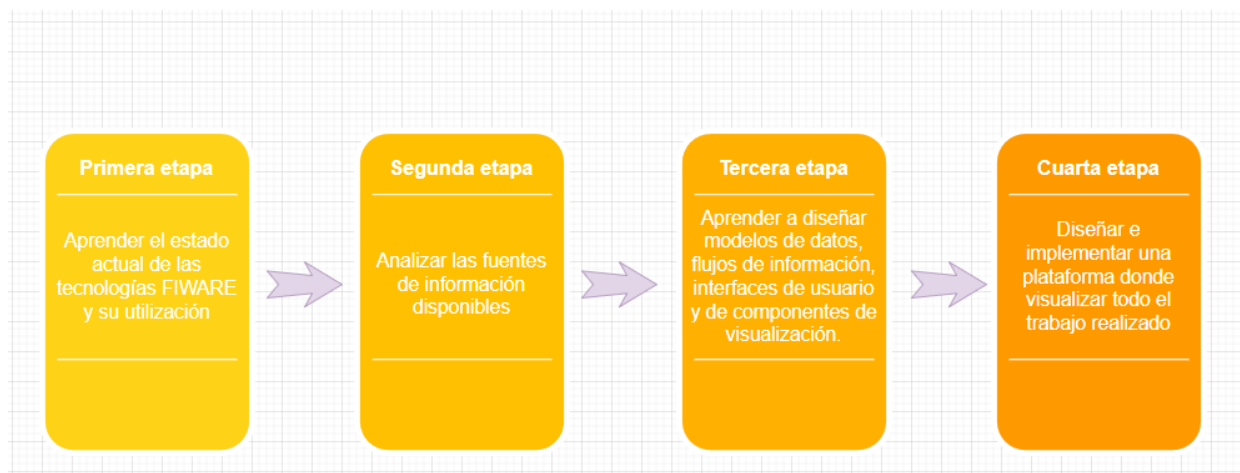
Las componentes tecnológicas que conforman este proyecto son las tecnologías Fiware, Google Cloud Platform y las herramientas de transformación y visualización de datos PowerBi y QGIS.

## 1.2 Objetivos

El **objetivo general** de este proyecto es la elaboración del diseño y prototipado de una plataforma orientada a la visualización de datos sobre residuos urbanos, que pueda ayudar y mejorar su gestión.

Centrándonos en los **objetivos específicos** del proyecto:

- Aprender sobre el estado actual de las tecnologías Fiware y su utilización.
- Analizar las fuentes de información, datos disponibles, así como de los requisitos del sistema de gestión inteligente de residuos a desarrollar.
- Aprender a diseñar modelos de datos, flujos de información, interfaces de usuario y de componentes de visualización.
- Diseñar e implementar una plataforma donde visualizar todo el trabajo realizado.



**Figura 1.1:** Etapas de trabajo

## 1.3 Desarrollo y tecnologías

La primera parte del proyecto consiste en, una vez fijado el marco de desarrollo del mismo, el estudio del estado actual de las tecnologías Fiware y su utilización, desde los servicios disponibles hasta los modelos de datos existentes.

Una vez realizado este estudio, se desarrollan procedimientos para la extracción, almacenamiento y transformación de datos públicos mediante diversas tecnologías. La extracción de datos abiertos y su transformación se realiza mediante scripts implementados con Python y escritos utilizando Visual Studio Code. Los datos utilizados son públicos, provenientes de fuentes abiertas como el ISTAC o Datos Abiertos de Canarias.

El almacenamiento de la información se realiza, por un lado, en Google Cloud Platform, donde se usan los módulos Bigquery, Compute Engine, y Cloud Storage; por otro lado, se usa Cygnus, un habilitador Fiware que permite guardar histórico de lo extraído en su plataforma. En Google Cloud Platform almacenamos datos provenientes de portales de datos abiertos y datos producto de las transformaciones realizadas. Cygnus, almacena información de la sensorización y de fuentes privadas, como la proporcionadas por Administraciones Locales y empresas de gestión de residuos. Toda la información manejada por Cygnus se almacena en el sistema de gestión de bases de datos relacional, MySQL . Los datos gestionados en Google Cloud Platform se almacenarán en formato de tablas Sql con BigQuery.

Para la visualización se utiliza la herramienta PowerBi, con ella se elaboran cuadros de mando. Los cuadros de mando están formados por indicadores, tablas y gráficas que permiten mostrar toda la información que se ha extraído, almacenado y procesado, presentada de una manera sencilla para el usuario final. Powerbi puede utilizar los datos desde MySQL y Bigquery, por lo que se puede utilizar tanto los datos almacenados en el GCP como los almacenados en Fiware.

También es útil la integración del Sistema de Información Geográfica QGIS con la plataforma Fiware. Mucha de la información disponible y de interés para el proyecto es y está georreferenciada. Esto permite la elaboración de mapas y realización de análisis, como por ejemplo: establecer la localización de elementos del sistema de recogida de residuos, como contenedores, papeleras o camiones; cálculos de distancias de rutas teniendo en cuenta la tipología de las calles o nivel de llenado de los contenedores, análisis demográfico asociado a la recogida, etc.

Por último, se desarrolla una portal web con Wordpress, para integrar la información y los procesos, accesible al usuario final donde mostrar toda la información elaborada mediante gráficas y mapas .

Todos estos procedimientos, tecnologías e información conforman la plataforma de inteligencia de datos gestión de residuos propuesta.

## 1.4 Organización de la memoria

El contenido de esta memoria, además de esta introducción, consta de cinco capítulos principales:

- **Capítulo 2: Fundamentos.** En este capítulo se recoge el estado del arte de los principales conceptos que se usan en el trabajo.
- **Capítulo 3: Tecnologías Fiware.** Capítulo cuyo contenido describe un resumen de las tecnologías Fiware, sus características y componentes .
- **Capítulo 4: Diseño y desarrollo de la plataforma orientada a datos para la gestión de residuos.** Este capítulo explica detalladamente las tecnologías utilizadas y los procedimientos de cómo se han utilizado. Además, explica un esquema de la arquitectura, así como una muestra como quedaría el diseño final de la interfaz de usuario de la plataforma final (Mockup).
- **Capítulo 5: Conclusiones y líneas futuras:** El capítulo final resume el trabajo realizado, extrayendo algunas conclusiones, así como las posibilidades existentes de continuar con el mismo.

# Capítulo 2 : Fundamentos

## 2.1 La importancia de los datos

Las tecnologías avanzan y con ella, la economía y otros ámbitos de las ciudades, creando entornos más competitivos y sostenibles de cara al futuro. En la actualidad, los datos y la información se han convertido en unos de los bienes más preciados por cualquier organización, destacando cada vez más la importancia en la calidad de los mismos como activo organizacional. [\[15\]](#)

Los datos se limitan a describir hechos u objetos, mientras que la información está compuesta por el conjunto de estos datos que dentro de determinado entorno sirven a los usuarios, empresas o entidades para la toma de decisiones. El uso de los datos para apoyar la toma de decisiones hace que las empresas aumenten su valor estratégico, pues un uso inteligente y adecuado de los datos proporciona ventajas competitivas.

En la actualidad, se puede observar como aumentan las organizaciones que están transformando sus productos y la prestación de servicios, con el uso de los datos que poseen, con el objetivo de convertirse en entidades más inteligentes. Así también las administraciones públicas, pasan de ser administraciones pasivas, rígidas y burocráticas a convertirse en administraciones más modernas, transparentes, proactivas y ágiles.

En una sociedad del conocimiento, de la economía del dato, las administraciones públicas necesitan una reestructuración y redefinición, mediante el uso inteligente de la información que poseen. La transformación digital de las organizaciones debe utilizar los recursos y fortalezas de estas para adaptar y ofrecer servicios que respondan a las necesidades y demandas actuales. Un nuevo modelo basado en la digitalización y en la gestión inteligente de los datos, donde las TIC juegan un papel central en el soporte de los servicios y en la toma de decisiones.

Los datos son el componente principal dentro de toda esta transformación. Todo el desarrollo y modernización de las organizaciones depende del acceso y uso de los mismos, pues estos aportan un conocimiento que es esencial para orientar y optimizar la gestión. Es más, a las administraciones esto les permite optimizar los servicios públicos, reducir costes y formular nuevas políticas y estrategias.



## 2.2 Inteligencia de Negocios

De la importancia de la información y la toma de decisiones inteligentes nace el **Inteligencia de Negocios o Business Intelligence (BI)**. Se define BI como el proceso analítico, con soporte tecnológico, que reúne y transforma datos fragmentados de empresas y mercados en información o conocimiento sobre objetivos, oportunidades y posiciones de una organización. [20]

Gartner (empresa consultora y de investigación en tecnologías de la información, con sede en Estados Unidos) en 1996, definió en uno de sus informes que *“las herramientas de reporte, consulta y análisis de datos pueden ayudar a los usuarios de negocios a navegar a través de un mar de información para sintetizar la información valiosa que en él se encuentra”*. [5] Este conjunto de estrategias y herramientas se le conoce como Inteligencia de Negocios (Inteligencia de Negocios), las cuales poseen una serie de características comunes como: la información que utilizan debe de ser accesible para cualquier usuario de la empresa, pues los datos son la base principal de todo este proceso; se apoya al usuario en la toma de decisiones, proporcionándole utilidades que le permitan seleccionar y manipular los datos que desee, y permitiéndole realizar análisis más personalizados que dan una mejor respuesta; y se utiliza un sistema de fácil manejo e intuitivo donde no sea de vital importancia los conocimientos técnicos del usuario final.

Utilizando el BI se obtienen beneficios que se clasifican en tangibles, intangibles y estratégicos. Ejemplo de beneficios tangibles son la generación de ingresos o reducción de costes. En cuanto a los intangibles, el hecho de que la información se encuentre disponible para sus usuarios permite que la utilicen para tomar decisiones y así mejorar la posición competitiva. Por último, los estratégicos, constituidos por todos aquellos que facilitan la elaboración de estrategias, indicando por ejemplo el mercado o clientes al que orientarse.

El proceso de utilización de técnicas y herramientas para la toma de decisiones, obteniendo beneficios para la organización, se produce gracias a la disposición de una serie de componentes básicos, que interactúan entre sí y que conforman la llamada, **Inteligencia de Negocio:**

1. Un problema al que queremos dar respuesta.
2. Un equipo de trabajadores que lleven a cabo los análisis.
3. Una base de datos accesible por los usuarios que contenga toda la información posible que sirva de utilidad (DataWarehouse).
4. Una aplicación de Inteligencia Empresarial que permita trabajar con la información disponible y visualizarla.

## 2.3 Sistemas de Información

Los **Sistemas de Información** (SI) se define como: “conjunto formal de procesos que, operando sobre una colección de datos estructurada de acuerdo a las necesidades de la empresa, recopila, elabora y distribuyen selectivamente la información necesaria para la operación de dicha empresa y para las actividades de dirección y control correspondientes, apoyando, al menos en parte, los procesos de toma de decisiones necesarios para desempeñar funciones de negocio de la empresa de acuerdo con su estrategia” [7].

Dentro de las tareas que realizan los SI en una organización tenemos cuatro principales que tratan con los datos: la entrada de datos, relacionados con clientes, pagos, facturas, etc; el procesamiento de los mismos, como el cálculo del saldo de un cliente o de sus intereses moratorios; el almacenamiento, por ejemplo, el catálogo de clientes o conjuntos de facturas; y la salida de resultados, tales como informes de pagos o consultas de saldo a través de una pantalla.

A partir de estas actividades básicas, los Sistemas de Información permiten la automatización de procesos operativos y proporcionan información a los usuarios que sirve de ayuda para la toma de decisiones.

Esta información puede provenir de diversas fuentes:

- Sistemas de información departamentales, como presupuestos, hojas de cálculo...
- Información externa, normalmente comprada a terceros.
- Sistemas operacionales o de transacción que incluyen aplicaciones desarrolladas a medida, ERP, SCM...

Los SI en la actualidad tienen grandes desafíos que abordar. A mayor cantidad de fuentes de información, más se complica su procesamiento y más capacidad de aprovechar la información existe para generar conocimiento. Si se accede a distintas fuentes, si la información está almacenada en bases de datos con diferentes sintaxis, el proceso de extracción y procesamiento de la información puede ser más complejo. Además, una vez seleccionadas las fuentes para la extracción de información, hay que transformarla o prepararla adecuadamente y entre otras, tener en cuenta la calidad de la misma. Los sistemas de Inteligencia de Negocios en este contexto vienen a reforzar los Sistemas de Información, aportando una capa de elaboración que facilite la generación de conocimiento y la transformación del mismo para la toma de decisiones.

## 2.4 Ciudades Inteligentes

El término de **Ciudad Inteligente (Smart Cities)**, según la AENOR (Asociación Española de Normalización y Certificación), hace referencia a una ciudad justa y equitativa centrada en el ciudadano que mejora continuamente su sostenibilidad y resiliencia aprovechando el conocimiento y los recursos disponibles, especialmente las Tecnologías y Comunicación (TIC), para mejorar la calidad de vida, la eficiencia de los servicios urbanos, la innovación y la competitividad sin comprometer las necesidades futuras en aspectos económicos, de gobernanza, sociales y medioambientales [\[1\]](#).

Las claves que convierten a una ciudad en una ciudad inteligente son [\[10\]](#):

- Empleo de las TIC.
- Automatización y control de edificios.
- Planificación urbana eficiente.
- Movilidad urbana y transporte público sostenible.
- **Gestión inteligente de residuos sólidos.**
- Mejora la sostenibilidad medioambiental.
- Preocupación por el entorno social.
- Tecnologías aplicadas a la educación y a la salud.
- Sistema de comercio electrónico.
- Máxima transparencia entre el gobierno y sus ciudadanos.
- Datos compartidos (Open Data).

En Canarias es también de interés replicar este paradigma de ciudad inteligente en todos los territorios, y como no podía ser de otra manera en territorios insulares trasladar el concepto de ciudades inteligentes al de islas inteligentes. La finalidad de los proyectos que conducen a este concepto es tener islas más autónomas y conectadas, que favorezcan y mejoren la calidad de vida de sus habitantes y el desarrollo sostenible, con el soporte de la innovación tecnológica y las tecnologías de la información y las comunicaciones.

En particular, Tenerife se encuentra inmersa en una estrategia que pretende preparar la isla de cara a los retos futuros de la modernización y la transformación digital, donde la tecnología es uno de los ejes principales, buscando siempre crear una mejora en la vida de sus habitantes.

Por otro lado, existe este mismo planteamiento trasladado a territorios donde la principal actividad es el turismo. Así en España y Canarias, se están impulsando proyectos de destinos turísticos inteligentes. Un **Destino Turístico Inteligente** es un destino innovador, consolidado sobre una infraestructura tecnológica de vanguardia, que garantiza el desarrollo sostenible del territorio, promueve la accesibilidad universal, facilita la

interacción e integración del visitante con el entorno incrementando su experiencia en el destino a la vez que mejora la calidad de vida de los residentes; comprometiéndose con el cumplimiento de los ODS (objetivos de desarrollo sostenible) de las Naciones Unidas [\[17\]](#).

Muchos municipios del archipiélago y de la isla de Tenerife también se han comprometido a convertirse en más transparentes, conectados y tecnológicos, centrados en las personas que residen en él. Adeje es uno de estos municipios, el cual mediante su proyecto Adeje Destino Turístico Inteligente, en el que se han invertido seis millones de euros, pretende brindar una atención de mayor calidad al usuario, tanto de cara al turista como a los propios habitantes del municipio.

Una parte de estos fondos está destinada dentro del municipio a mejorar la **gestión de los residuos**, lo cuál es de interés en el desarrollo de este TFG.

## 2.5 Gestión de residuos

Según Residuos Profesional [\[16\]](#), una revista profesional digital orientada al medio ambiente, gestión de residuos y reciclaje, en las ciudades se genera el 50% de los residuos globales del planeta y que estos generan entre el 60%- 80% de los gases de efecto invernadero. Además, este problema se vuelve cada día aún mayor debido a la grave sobrepoblación a la que se enfrenta nuestro planeta en las últimas décadas. A mayor cantidad de gente, mayor basura se genera, por lo que es de vital importancia crear cuanto antes sistemas que den solución o que por lo menos reduzcan esta problemática.

La gestión más eficiente de los restos y la mejora de los servicios urbanos de recogida son retos importantes a los que se enfrentan las ciudades. La gestión de residuos es un pilar fundamental de la economía circular, de las ciudades y destinos turísticos inteligentes. La gestión de residuos urbanos se compone de distintas etapas, desde la recolección y el transporte hasta su tratamiento. La recolección de residuos sólidos es una actividad clave del sistema de gestión para mejorar el servicio al ciudadano y también en la reducción de impactos medioambientales, así como del objetivo de conseguir un ahorro de costes para la administración.

Los municipios para la gestión de residuos sólidos necesitan sistemas integrados de información. Para ofrecer estos servicios municipales en las mejores condiciones, necesitan la optimización de los recursos públicos dispuestos. Un sistema integrado de gestión de residuos, con el diseño de estrategias y modelos adecuados, puede mejorar la planificación de rutas, la recogida, el mantenimiento y la gestión de los vehículos. La gestión inteligente de los residuos necesita de la ayuda de sistemas inteligentes que proporcionen las herramientas adecuadas de información y toma de decisiones.

Para el desarrollo de modelos de evaluación de las estrategias de recogida, la ubicación de depósitos y el establecimiento de rutas adecuadas en tiempo y espacio, es importante contar con los datos primarios recopilados de fuentes diversas y de la principal, del propio sistema de gestión, además de contar con herramientas adecuadas capaces de almacenar, recuperar, analizar, planificar y aplicar gran cantidad de datos, así como visualizar la salida con un tiempo de respuesta razonable. El conocimiento generado por los sistemas de gestión de residuos, analizados con técnicas y metodologías adecuadas, son de gran interés para mejorar la gestión y para la elaboración de nuevas estrategias y medidas. Los sistemas de información, la Inteligencia de Negocios y las tecnologías emergentes son herramientas fundamentales para conseguir estos propósitos.

Con la aplicación de tecnologías Fiware se podrá capturar información en tiempo real de sensores, disponer de históricos de información registrada, realizar estadísticas e indicadores para ver características concretas del entorno, transformar y elaborar la información obtenida para establecer actuaciones en consecuencia. Además, asociado a los objetivos del TFG podríamos determinar las características generales de interés del sistema de gestión de residuos y evaluar el desempeño orientado a su mejora, aumentar el reciclaje o la calidad de los servicios, así como encontrar, estimar y evaluar escenarios, patrones de comportamiento y predicción que faciliten la adecuación de los recursos y servicios de gestión dispuestos.

# Capítulo 3 : Tecnologías Fiware

## 3.1 Fiware

**Fiware** [6] es una plataforma, impulsada por la UE, para el desarrollo y despliegue de aplicaciones en Ciudades Inteligentes. El objetivo del conjunto de tecnologías que la integran es crear soluciones inteligentes a los diversos retos que presentan las ciudades. Estas soluciones tienen como particularidad que recogen información del entorno en el que se desenvuelven desde distintas fuentes, tratando de obtener una respuesta lo más eficiente y adecuada posible para la resolución de problemas. A la información recogida del entorno se le conoce en Fiware como **información de contexto**.



**Figura 3.1** : Logo Fiware

Fiware brinda microservicios basados en tecnologías open source, cuyo objetivo principal es realizar una mejora en la gestión de los datos, a través de la creación de estándares que faciliten su intercambio y armonización. Para esto último, Fiware dispone y facilita la creación de modelos de datos específicos y posee su propio Data MarketPlace (espacio compartidos de datos y modelos), que permiten el intercambio de información.

Fiware pretende construir una comunidad abierta e independiente donde todos sus miembros tengan la misma visión, construir un ecosistema sostenible y abierto de herramientas y aplicaciones para todos. La finalidad y el alcance de esto es facilitar el desarrollo de aplicaciones inteligentes en todos los sectores y ámbitos de aplicación.

Esta comunidad está formada por los desarrolladores de las tecnologías Fiware, y también de todos aquellos que contribuyen a la expansión de su ecosistema y lo hacen sostenible. La transparencia, apertura e independencia en la toma de decisiones son los pilares fundamentales de la comunidad Fiware.



**Figura 3.2:** Empresas miembro de la comunidad Fiware

Fiware es una plataforma que está extendiendo su uso y aplicación. Son muchas las ciudades y territorios, empresas y emprendedores que la utilizan en la creación de aplicaciones abiertas. Estas aplicaciones implementan modelos de plataformas colaborativas, donde la información es compartida por todos los actores que intervienen en la misma y consultada por todos. Atos o Telefónica son dos grandes empresas que están detrás del uso de esta tecnología y que la implantan en sus proyectos.

## 3.2 Arquitectura y componentes

En esta sección vamos a describir las componentes más importantes y la arquitectura de Fiware, haciendo hincapié en aquellas que son de interés para nuestro proyecto.

El **Context Broker** [9], por defecto el Orion Context Broker, es el único componente obligatorio y necesario para el desarrollo de aplicaciones con Fiware. Se trata de un servidor que permite manejar la **información de contexto**, utilizando peticiones desde una API tipo REST.

Con el Context Broker se cumple con uno de los principios fundamentales de Fiware, el que consumidores y productores de información de contexto están separados y disociados. Esto quiere decir que, ninguno conoce al otro, el productor no conoce quien va a recibir los datos y el consumidor no conoce quién los produce. A ambos les basta únicamente estar conectados a la plataforma Fiware.

Toda la información de contexto que maneja el Context Broker es generada por los productores de contexto, sistemas de información a los que se conecta, sensores o cualquier dispositivo que capte información y la envíe a Fiware para que esta sea accesible para los consumidores. Estos son consumidores de contexto, son los interesados en procesar, analizar y utilizar toda esa información disponible.

Tanto el Context Broker como el resto de servicios (habilitadores), se instalan mediante el uso de Docker, una herramienta de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software.

```

cris11an@perisistao-virtualBox:~/desktop/calidadatire$ sudo docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
ae514f5d2130   smartsdk/quantumleap               "python app.py"         30 seconds ago Up 28 seconds 0.0.0.0:8668->8668/tcp, :::8668->8668/tcp
f0b4cda14db4   fiware/orion:1.13.0                "/usr/bin/contextBro..." 34 seconds ago Up 30 seconds (health: starting) 0.0.0.0:1026->1026/tcp, :::1026->1026/tcp
c45353f035a3   mongo:3.2                           "docker-entrypoint.s..." 38 seconds ago Up 34 seconds 0.0.0.0:27017->27017/tcp, :::27017->27017/tcp
a18d6fa1aa64   crate:1.0.5                         "docker-entrypoint.s..." 38 seconds ago Up 34 seconds 0.0.0.0:4200->4200/tcp, :::4200->4200/tcp, 0.0.0.0:4300->4300/tcp, :::4300->4300/tcp
32-5532/tcp    calidadatire_crate_1

```

Figura 3.3: Servicios Fiware levantados en Docker

El Context Broker se instala asociado al puerto 1026 de los equipos, utilizando MongoDB (puerto 27017) como almacén de la información de contexto.

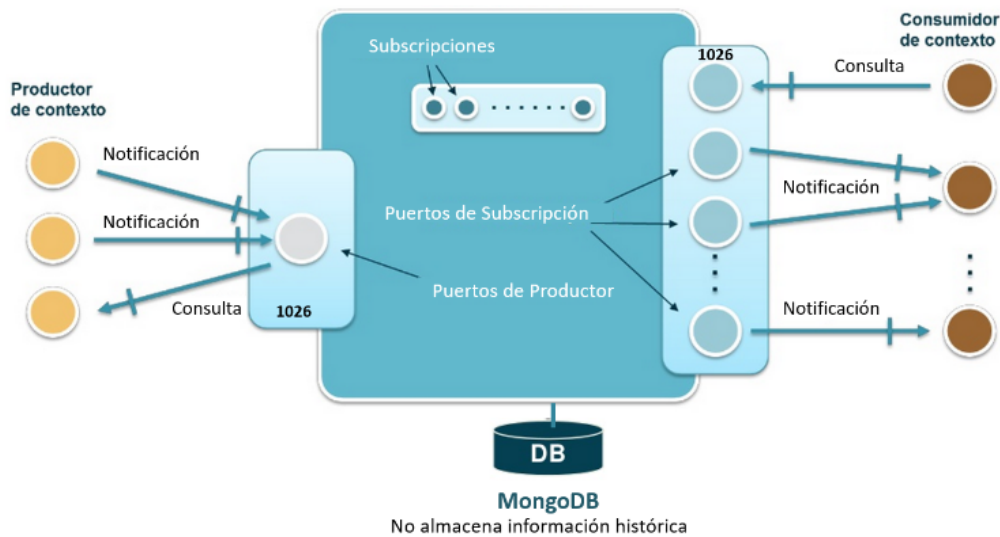
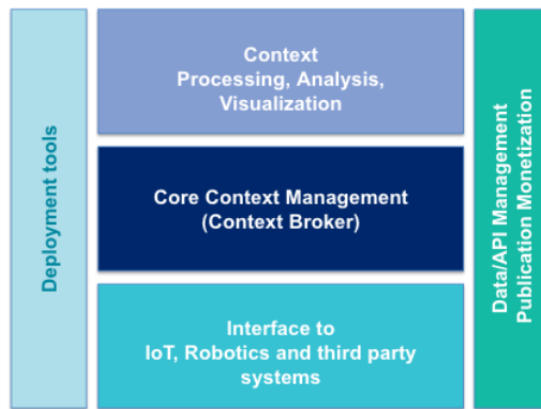


Figura 3.4: Arquitectura y funcionamiento del Context Broker

Sobre el Context Broker, existen una serie de módulos open source de Fiware que cumplen diversas funciones como el análisis o visualización de la información de contexto, ayudando así a los usuarios finales en la toma de decisiones inteligentes. Toda la comunicación entre los distintos módulos utilizados en Fiware, con el Context Broker como componente central, se lleva a cabo mediante una **API RESTful NGSI**.





**Figura 3.5:** Arquitectura de la plataforma Fiware

Los módulos opcionales de Fiware que se pueden integrar con el Context Broker se clasifican por categorías en función de la labor que realizan:

<p><b>IoT, Robots Y Sistemas De Terceros</b></p>	<p>Habilitadores que facilitan la conexión con el Internet de las cosas, robots y sistemas de terceros con el fin de recopilar información de contexto importante, desencadenando actuaciones en respuesta. IDAS o OpenMTC son ejemplos de estos habilitadores</p>
<p><b>Procesamiento, Análisis Y Visualización</b></p>	<p>Habilitadores que facilitan el procesamiento, análisis y la visualización de la información de contexto con el objetivo de implementar el deseado “comportamiento inteligente” en una aplicación. Wirecloud es un ejemplo de estos habilitadores, permitiendo crear dashboards altamente personalizables de cara al usuario final</p>
<p><b>Gestión De Acceso, Publicación Y Monetización</b></p>	<p>Habilitadores para la publicación y monetización de la información de contexto manejada a través del Context Broker de la plataforma Fiware. Bliz Framework es un ejemplo de estos habilitadores, dando soporte de back-end a la monetización de datos/APIs de contexto.</p>

**Tabla 3.1:** Habilitadores opcionales de Fiware

### 3.3 Modelos de datos

Con el objetivo principal de adoptar una arquitectura de referencia, y el uso de estándares donde almacenar la información de contexto, se han desarrollado modelos de datos comunes [11] que generan un mercado digital de soluciones inteligentes, utilizables en distintos sectores. Estos modelos de datos deben cumplir con las características de ser libres uso, modificación e intercambio.

Están agrupados en temáticas, donde cada una de estas tiene un repositorio en Github. A estas temáticas se les conoce también como dominios, que representan distintos ámbitos dentro de la industria. Actualmente en el sitio web oficial de Fiware se definen doce dominios distintos, cada uno con su repositorio, que a su vez está constituido por diversos modelos de datos. A continuación presentamos en la siguiente tabla, un resumen de los diferentes dominios.

<b>Smart Agrifood</b>	Dominio orientado a la optimización de la producción en granjas, utilizando las tecnologías del momento manteniendo siempre la sostenibilidad
<b>Smart Cities</b>	Dominio orientado en la utilización de las tecnologías Fiware para la creación y desarrollo de las Smart Cities. ya definidas anteriormente.
<b>Smart Energy</b>	Dominio orientado a la creación y optimización de modelos de administración de la energía
<b>Smart Environment</b>	Dominio que utiliza modelos de datos referentes al medio ambiente.
<b>Smart Sensoring</b>	Dominio que utiliza modelos de datos referentes al uso de sensorización
<b>Smart Water</b>	Dominio relacionado con la utilización de modelos de datos para la optimización en la administración del agua.
<b>Smart Destination</b>	Dominio relacionado con modelos de datos referentes a destinos turísticos inteligentes.
<b>Smart Manufacturing</b>	Dominio relacionado con el sector manufacturero.

<b>Smart Health</b>	Dominio relacionado con el sector de la salud. Utilizado durante la pandemia del Covid-19 para almacén de datos sobre vacunación.
<b>Smart Aeronautics</b>	Dominio relacionado con el sector aeronáutico.
<b>Smart Robotics</b>	Dominio orientado a todo lo relacionado con el mundo de la robótica.
<b>Cross Sector</b>	Dominio que contiene todos los modelos de datos que se pueden utilizar en distintos dominios

**Tabla 3.2:** Dominios de datos de Fiware

Todos los modelos de datos que forman parte de cada uno de estos dominios están divididos en entidades. Cada entidad hace referencia a un campo de información distinto dentro del modelo, que a su vez posee una serie de atributos diferenciados. Estas entidades pueden repetirse en distintos modelos de datos.

### 3.4 Casos de uso

Actualmente en España existen varias ciudades que utilizan las tecnologías Fiware con el propósito principal de dar soporte a las ciudades inteligentes. Por ejemplo, [SmartCity Valencia](#) es el proyecto de plataforma de servicios de la ciudad de Valencia. En este proyecto Valencia participa junto con otras treinta ciudades de siete países distintos que apuestan por Fiware como estándar para el desarrollo de ciudades inteligentes. Dentro de España, Santander, Málaga y Sevilla son otras de las ciudades que se han sumado a esta iniciativa.

Un caso de interés para este proyecto, es el del ayuntamiento de Badajoz, que está llevando a cabo un proyecto utilizando tecnologías Fiware bajo el nombre de [Badajoz es más](#). Uno de los ámbitos en los que se centra el proyecto es la gestión de residuos. El objetivo del sistema inteligente desarrollado en este ámbito es mejorar la gestión de residuos en pequeñas localidades, en las que el servicio de recogida de basuras a menudo, está compartido entre varios municipios. Mediante sensores el sistema es capaz de detectar el nivel de llenado de cada contenedor y puede obtener información actualizada de cada punto de recogida, pudiendo ajustar las rutas de recogida para una mayor eficiencia.



**Figura 3.6:** Visualización de datos de Badajoz es más

En conclusión, se puede afirmar que Fiware es una herramienta que se encuentra en crecimiento. Cada vez más ciudades la utilizan bajo el paradigma de la creación de territorios, ciudades y destinos inteligentes. Por ello, las tecnologías Fiware, constituyen uno de los pilares fundamentales en el proyecto que estamos realizando de creación de una plataforma de gestión de residuos.

# Capítulo 4 : Diseño y desarrollo de la plataforma orientada a datos para la gestión de residuos

En este capítulo se describe el diseño propuesto de la plataforma, su prototipado e implementación de algunas de las tecnologías incorporadas. Así, se propone una arquitectura de integración de distintos componentes con herramientas que ayudan a la captura, transformación, almacenamiento y visualización de datos. Una arquitectura que, además, es interoperable e integrada con la plataforma Fiware.

La captura y transformación de datos se realiza, por un lado, con scripts de extracción de datos abiertos desarrollados en Python y, por el otro, por la plataforma Fiware, encargada de recoger información de sensorización y de la proporcionada por las administraciones y empresas de gestión de residuos.

El almacenamiento de la información proveniente de los portales abiertos es realizado por Google Cloud Platform y, por otro lado, Fiware almacena la información de sensorización y la proveniente de las administraciones locales y de las empresas.

PowerBi se encarga de la visualización de la información y conocimientos elaborados y transformados. QGIS se utiliza con los datos georeferenciados disponibles en Fiware para la elaboración de mapas, geolocalización, análisis espacial y otros procedimientos. Toda esta parte que podemos denominar visualización de datos, es accesible y se podrá utilizar por el usuario final mediante un portal web desarrollado con Wordpress.

## 4.1 Arquitectura

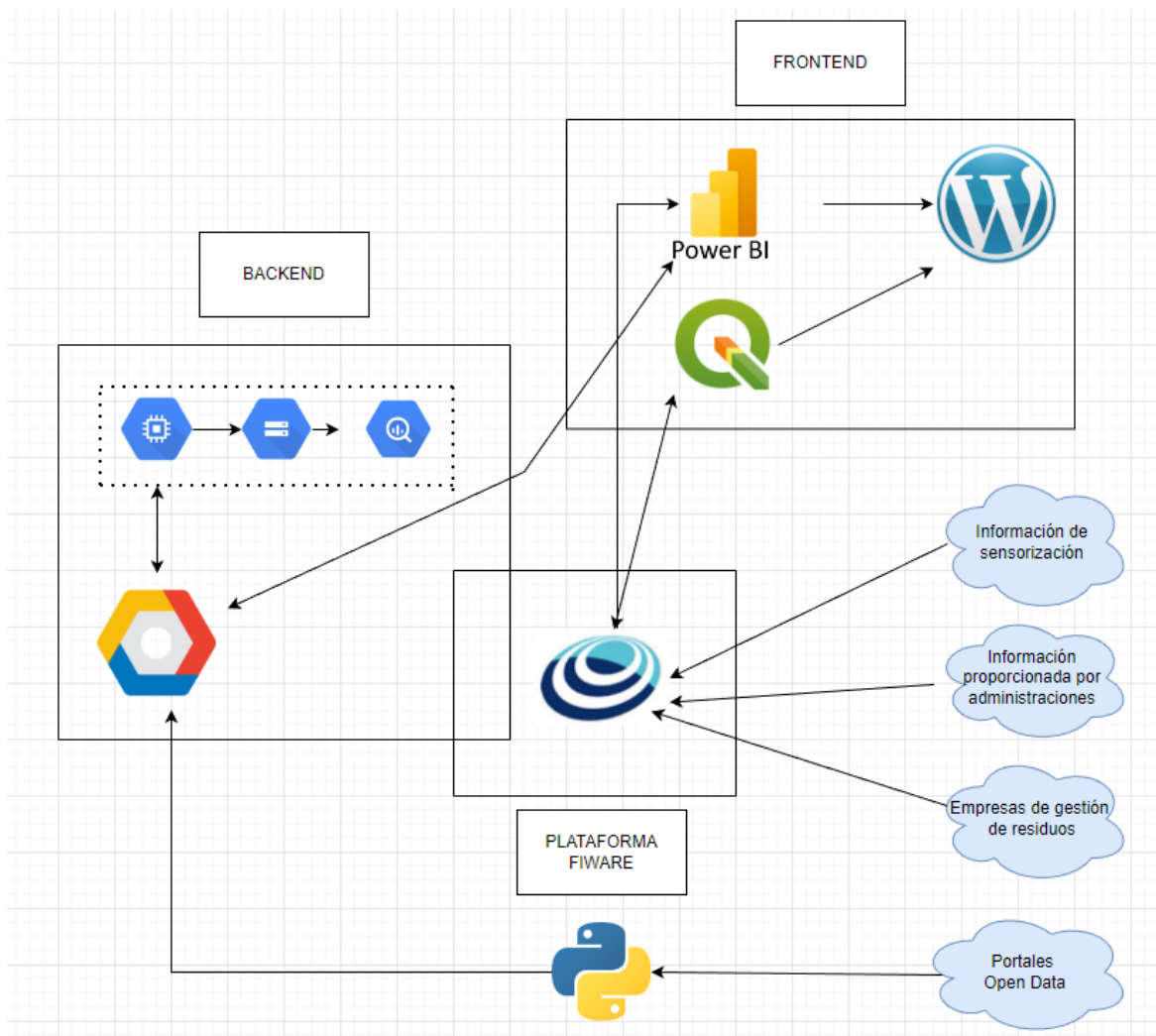
La arquitectura de este proyecto está estructurada en dos bloques: **Frontend** y **Backend**. El primero es la parte de la plataforma formada por herramientas que facilitan la interacción al usuario final, mientras que el segundo incorpora todas las tecnologías que capturan, almacenan y procesan lo que va a alimentar el Frontend de información.

El Backend del proyecto está constituido por las tecnologías que forman parte de dos plataformas: Fiware y Google Cloud Platform. La primera estará destinada al manejo de toda la información proveniente de la sensorización así como de las administraciones y empresas de gestión de residuos. La segunda, actuará como centro de almacenamiento de la información extraída de portales abiertos. Esta información, proveniente de los

portales abiertos, es extraída y almacenada en Google Cloud Platform mediante scripts escritos en Python.

El Frontend está compuesto por tres herramientas principales:

- PowerBi: que permite la elaboración de gráficas y cuadros de mando.
- QGIS: mediante el cuál se elaboran mapas por capas donde se mostrará, por ejemplo, la ubicación de cubos y contenedores así como datos relacionados con los mismos. Toda esta información proviene de la plataforma Fiware.
- Wordpress: con el que se desarrolla una página web destinada al usuario final.



**Figura 4.1:** Diseño de la arquitectura de la plataforma

Se utilizan dos tipos de fuentes de datos de las que extrae la información que se muestra en la plataforma:

- Fuentes de datos abiertas (ISTAC y Datos Abiertos Canarias).
- Información privativa que proviene de sistemas de información de las administraciones locales, empresas privadas que participan en el proyecto o dispositivos sensorizados.

Para los datos provenientes de portales de open data, su extracción, procesado y transformación se hace mediante unos scripts desarrollados en Python. Para la ejecución de estos se utiliza el servicio de Compute Engine, que permite crear máquinas virtuales y ejecutar en las mismas los scripts de forma periódica y automática (mediante la creación de "Crons"). Esto se debe a que al utilizar fuentes abiertas que se van actualizando, es de vital importancia establecer unos periodos para la ejecución de los scripts para que la información extraída siempre sea la más reciente.

Una vez extraída y procesada la información, se almacena mediante el servicio de Cloud Storage para posteriormente ser pasadas a BigQuery, que actuará a modo de data warehouse, donde creamos vistas que utilizaremos con Powerbi para desarrollar los cuadros de mando. Estos cuadros de mando se muestran embebidos en nuestra página web final.

Para los datos privados provenientes de la sensorización, administraciones y empresas de gestión de residuos, se utilizará la plataforma Fiware para su captación y el módulo Cygnus para su almacenamiento. Este módulo permite almacenar la información captada en bases de datos en MySQL. Esta información será utilizada por dos herramientas distintas: QGIS, para elaborar mapas con las localizaciones de, por ejemplo, los contenedores, cubos y camiones de la ciudad; y Powerbi, para la elaboración de gráficas.

En resumen, con la información pública y privada almacenada se elaboran cuadros de mando y con la información de contexto generada por los sensores y gestionada por Fiware, se muestran mapas con la ubicación de, por ejemplo, los distintos contenedores, cubos y camiones de la ciudad. Toda esta parte destinada a la visualización de la información estará montada en un portal web creado con Wordpress.

## 4.2 Integración de componentes tecnológicas




### 4.2.1 Google Cloud Platform

**Google Cloud Platform** es una plataforma en la nube de Google que nos proporciona servicios de distinta índole que permiten cumplir con las tareas necesarias dentro del proyecto para que este funcione correctamente.



**Figura 4.2:** Logo Google Cloud Platform

Para este proyecto se han utilizado tres de estos servicios: Compute Engine, Bigquery y Cloud Storage.

	Compute Engine <a href="#">[4]</a> nos permitirá ejecutar de manera automática y periódica los scripts de extracción de los datos abiertos para que la información siempre sea la más reciente.
	Cloud Storage <a href="#">[3]</a> nos proporciona un espacio en la nube donde almacenar todos esos datos obtenidos, extraídos de fuentes públicas.
	BigQuery <a href="#">[2]</a> , que nos permite transformar la información y almacenarla en forma de bases de datos ,lo cuál facilita su uso de cara a la su vista mediante herramientas de visualización (Powerbi)

**Tabla 4.1:** Servicios utilizados de Google Cloud Platform

La conexión a la plataforma en la nube de Google se realiza mediante un usuario, al cual se le adjudican unas credenciales (en formato .json). Durante el desarrollo de los scripts de extracción se indica cuales son estas credenciales y así se permite la interacción con



el GCP, dándonos la posibilidad de subir al Cloud Storage los datos extraídos.

## 4.2.2 Scripts de extracción

Para el desarrollo de los scripts de extracción se utiliza Python como lenguaje de programación y Visual Studio Code[18] como editor de código fuente. Las librerías principales de Python utilizadas son la de Cloud Storage [12] y la de Pandas [13]. La primera porque permite cargar datos en Google Storage y tenerlos almacenados en un lugar de manera centralizada para luego hacer uso de ellos. La segunda posibilita la extracción de los datos abiertos en formato .json ,su formateo y al almacenamiento en estructuras bidimensionales de datos llamadas dataframes, las cuáles facilitan posteriormente su tratado y visualización de forma gráfica. Estos scripts realizan procesos ETL, es decir, procesos de extracción, transformación y carga de datos.



Figura 4.3: Logo de Python

En primer lugar, se realiza un script de extracción de datos en crudo en formato .json de los datos y se almacenan en Cloud Storage.

```
uuids = [
    "e06ff7ff-b1e3-4a3a-90be-ad44c7ab2ec3",
    "4e195ab3-2a8d-4aab-b480-11070dfff7cc",
    "4b306c9a-6d96-435b-ae75-0f27173ef019",
    "2c66ebe1-4180-4fa6-b0e7-1e905b4e755b",
    "2a5d34a6-c97a-4c6c-b404-7b2e25061a92",
    "f49b8aea-fb28-4da4-926b-b7f58ac8cb23"
]
cm = 'Superficies'
for uuid in uuids:
    url = f'http://www.gobiernodecanarias.org/istac/jaxi-istac/tabla.do?accion=jsonMtd&uuidConsulta={uuid}'
    print(url)
    try:
        response = requests.request("GET", url)
        if response:
            exportToStorage(cm, json.dumps(response.json()), uuid)
        else:
            return "No data to show"
    except requests.exceptions.RequestException as e:
        raise SystemExit(e)
```

Figura 4.4: Script de extracción de datos en crudo (I)

```

def exportToStorage(cm, data, tabla):
    """
    Exporta la respuesta RAW a Google Storage
    """
    client = storage.Client(project = "proyecto de GCP")
    bucket = client.get_bucket('TFG_bucket')
    blob = bucket.blob(f'Medioambiente_EconomiaCircular/{cm}/raw/{tabla}.json')

    try:
        blob.upload_from_string(data)
        print('Copied' + tabla + '.json to Google Storage {0}'.format("Modified" if blob.exists() else "New"))
        return "Ok"
    except Exception as e:
        print(tabla + ".json does not exist {0}".format(e))
        return "Error"

```

**Figura 4.5:** Script de extracción de datos en crudo (II)

Para ello, la primera labor realizada en el script es almacenar todos los uids de los cubos de datos abiertos que se quieran consultar de [Datos Abiertos de Canarias](#). Luego, se descargan uno a uno para posteriormente almacenarlos en la nube de Google. Estos cubos conforman el conjunto de datos en **crudo**. Es de gran importancia indicar nuestras credenciales de acceso, para que así la carga de datos extraídos se realice de forma adecuada.

Estos cubos de información en crudo son posteriormente procesados por otro script y almacenados de forma ordenada en dataframes que facilitan su tratamiento posterior. Estos conjuntos de datos son los procesados y se encuentran en formato .csv.

```

def ETL_1701()->str:
    tabla = "e06ff7ff-b1e3-4a3a-90be-ad44c7ab2ec3"
    cm = "Superficies"

    json_array = readRaw(cm, tabla)

    df = pd.DataFrame(columns=['Isla', 'Municipio', 'Indicador', 'Valor'])

    origin_codes = [origin_code for origin_code in json_array['categories'][0]['codes']]
    origin_labels = [label for label in json_array['categories'][0]['labels']]
    origins = dict(zip(origin_codes, origin_labels))

    indicator_codes = [indicator_code for indicator_code in json_array['categories'][1]['codes']]
    indicator_labels = [label for label in json_array['categories'][1]['labels']]
    indicators = dict(zip(indicator_codes, indicator_labels))

    islands = {
        'Canarias': origin_labels.index('CANARIAS'),
        'Lanzarote': origin_labels.index(' LANZAROTE'),
        'Fuerteventura': origin_labels.index(' FUERTEVENTURA'),
        'Gran Canaria': origin_labels.index(' GRAN CANARIA'),
        'Tenerife': origin_labels.index(' TENERIFE'),
        'La Gomera': origin_labels.index(' LA GOMERA'),
        'La Palma': origin_labels.index(' LA PALMA'),
        'El Hierro': origin_labels.index(' EL HIERRO')
    }

    for item in json_array['data']:

        data = item
        origin = search_origin_by_code(data['dimCodes'][0], origins)
        origin_index = origin_labels.index(origin)
        indicator = search_indicator_by_code(data['dimCodes'][1], indicators)

        island = ''

        for isla in islands.items():
            if origin_index >= isla[1]:
                island = isla[0]
            else:
                break

```

**Figura 4.6:** Script de procesado de datos crudos(I)

```

value = 0 if data['Valor'] == '..' or data['Valor'] == '.' else data['Valor']
df = df.append({
    'Municipio': origin,
    'Isla': island,
    'Indicador': indicator,
    'Valor': value
}, ignore_index = True)

turists = df
print(df)
return turists

```

**Figura 4.7:** Script de procesado de datos crudos(II)

Tras este script, los datos se encuentran limpios y almacenados en Cloud Storage. El siguiente paso es pasar todos los datos procesados del formato .csv en el que están almacenados a un formato que pueda ser utilizado por un sistema de visualización de información. Para ello, se utiliza BigQuery, con el cuál se realizan se crean bases de datos

y se generan vistas de las mismas con la información que se posee en Cloud Storage, de manera que estas vistas sean las que se usan posteriormente para realizar la visualización de la información de manera más sencilla. Esto se debe a que PowerBi da la opción de escoger BigQuery como fuente de datos externa para la elaboración de las gráficas.

Resultados de la consulta

INFORMACIÓN DEL TRABAJO			RESULTADOS	JSON	DETALLES DE LA EJECUCIÓN					
Fila	Isla	Municipio	Indicador	Valor	Dimension	Proteccion	Destino	A_o	Frecuencia	Tipo_residuo
11601	Tenerife	null	null	100.0	Hogares y residuos	null	null	2018	TOTAL	Papel y cartón
11602	Tenerife	null	null	100.0	Hogares y residuos	null	null	2013	TOTAL	Papel y cartón
11603	Tenerife	null	null	100.0	Hogares y residuos	null	null	2007	TOTAL	Papel y cartón
11604	Tenerife	null	null	100.0	Hogares y residuos	null	null	2004	TOTAL	Papel y cartón
11605	Tenerife	null	null	100.0	Hogares y residuos	null	null	2018	TOTAL	Plásticos, tetrabrik y latas
11606	Tenerife	null	null	100.0	Hogares y residuos	null	null	2013	TOTAL	Plásticos, tetrabrik y latas
11607	Tenerife	null	null	100.0	Hogares y residuos	null	null	2007	TOTAL	Plásticos, tetrabrik y latas
11608	Tenerife	null	null	100.0	Hogares y residuos	null	null	2004	TOTAL	Plásticos, tetrabrik y latas
11609	Tenerife	null	null	100.0	Hogares y residuos	null	null	2018	TOTAL	Pilas
11610	Tenerife	null	null	100.0	Hogares y residuos	null	null	2013	TOTAL	Pilas
11611	Tenerife	null	null	100.0	Hogares y residuos	null	null	2007	TOTAL	Pilas
11612	Tenerife	null	null	100.0	Hogares y residuos	null	null	2004	TOTAL	Pilas
11613	Tenerife	null	null	100.0	Hogares y residuos	null	null	2018	TOTAL	Medicamentos
11614	Tenerife	null	null	100.0	Hogares y residuos	null	null	2013	TOTAL	Medicamentos
11615	Tenerife	null	null	100.0	Hogares y residuos	null	null	2007	TOTAL	Medicamentos
11616	Tenerife	null	null	0.0	Hogares y residuos	null	null	2004	TOTAL	Medicamentos
11617	Tenerife	null	null	19.38	Hogares y residuos	null	null	2018	Nunca	Cristal
11618	Tenerife	null	null	23.6	Hogares y residuos	null	null	2013	Nunca	Cristal

Figura 4.8: Datos en BigQuery

Con una mayor involucración por parte de las ciudades, se podría utilizar información más específica como fuentes de datos, en vez de limitarse a seleccionar la información del ISTAC o de Datos Abiertos de Canarias. Adeje, por ejemplo, va a habilitar una [web](#) de datos abiertos del municipio en un futuro cercano. Los scripts definidos en este apartado podrían adaptarse y ser utilizados para estas nuevas fuentes de información una vez estén disponibles, permitiendo posteriormente crear cuadros de mando mucho más específicos del municipio.

### 4.2.3 Modelo de datos y habilitadores Fiware utilizados

Para el desarrollo de esta plataforma los dominios que contienen los modelos de datos más útiles son: Smart Cities y Smart Sensoring. Estos modelos proporcionan la información en formato .json o .Geojson.

En el dominio Smart Cities, tenemos un modelo de datos referente la gestión de residuos, el [WasteManagement](#). Este modelo contiene entidades que suelen estar involucradas en

la retirada de residuos. Ha sido diseñado para que sea lo más general posible, de manera que su utilización en distintos ámbitos de la gestión de restos sea relativamente sencilla. Es de utilidad tanto para contenedores de superficie, como para los enterrados o incluso para los cubos de basura colocados en las calles para la retirada de pequeñas cantidades de residuos. Las tres entidades principales de este modelo son:

<b>WasteContainer</b>	Entidad que contiene información general sobre un contenedor
<b>WasteContainerIsle</b>	Entidad que contiene información general sobre un conjunto de contenedores(isla)
<b>WasteContainerModel</b>	Entidad con información modelo de contenedor de residuos que captura las propiedades estáticas de una clase de contenedores.

**Tabla 4.2:** Entidades del modelo de datos WasteManagement

A su vez, cada una de estas entidades posee una serie de etiquetas que servirán para identificar diversas características de un contenedor. Por ejemplo, en la entidad **WasterContainer**, tenemos una distribución de [etiquetas](#) donde destacan:

<b>id</b>	Identificador único de la entidad
<b>type</b>	Tipo de entidad NGSi: Tiene que ser WasteContainer
<b>location</b>	Referencia <b>Geojson</b> al elemento. Puede ser Point, LineString, Polygon, MultiPoint, MultiLineString o MultiPolygon
<b>binCapacity</b>	Capacidad total en términos de volumen de residuos que puede contener el contenedor.
<b>dateLastEmptying</b>	Marca de tiempo que representa cuándo se vació el contenedor por última vez.
<b>cargoWeight</b>	Peso de la carga del contenedor.

**Tabla 4.3:** Etiquetas de la entidad WasteContainer

Únicamente con estas etiquetas, el usuario podrá visualizar cuando fue la última vez que se vació el contenedor, cual es su carga máxima y cuál su carga actual. Con esta información y con la proporcionada por el resto de etiquetas del modelo, se pueden

establecer actuaciones y protocolos de funcionamiento en función a la situación de los contenedores en el momento. Un ejemplo de una entidad WasteContainer completa sería la siguiente:

```
{
  "id": "urn:ngsi-ld:wastecontainer:1021:AAWD",
  "type": "WasteContainer",
  "location": {
    "coordinates": [
      -8.768460000000001,
      42.60214472222222
    ],
    "type": "Point"
  },
  "binCapacity": 43,
  "binColor": "Green",
  "binClearedTime": "2021-03-11T15:51:02+05:30",
  "wardId": "21",
  "binCategory": "Household Bin",
  "license_plate": "KA23F2345",
  "RFID": "67855734",
  "binFillingLevel": 0.65,
  "binFullnessThreshold": 80,
  "binRecommendedLoad": 30,
  "binId": "12",
  "binMaxLoad": 75,
  "binLoggedTime": "2021-03-01T15:51:02+05:30"
}
```

**Figura 4.9:** Estructura de la entidad WasteContainer

Aparte del modelo WasteManagement, es de especial interés también el modelo [Device](#), del dominio Smart Sensoring. Este modelo también es de ayuda para la gestión de residuos, pues utiliza sensores que están constantemente monitorizados y la información que nos brinda para ayudarnos en nuestro trabajo.

Con este modelo se puede saber la localización de los sensores, su nivel de batería, fecha de instalación así como medir el nivel de llenado de los contenedores para saber cuánto les falta para alcanzar su capacidad máxima. Un ejemplo de la entidad Device es la siguiente:

```

{
  "id": "device-9845A",
  "type": "Device",
  "category": [
    "sensor"
  ],
  "controlledProperty": [
    "fillingLevel",
    "temperature"
  ],
  "controlledAsset": [
    "wastecontainer-Osuna-100"
  ],
  "ipAddress": [
    "192.14.56.78"
  ],
  "mcc": "214",
  "mnc": "07",
  "batteryLevel": 0.75,
  "serialNumber": "9845A",
  "refDeviceModel": "myDevice-wastecontainer-sensor-345",
  "rssi": 0.86,
  "value": "1%3D0.22%3Bt%3D21.2",
  "deviceState": "ok",
  "dateFirstUsed": "2014-09-11T11:00:00Z",
  "owner": [
    "http://person.org/leon"
  ]
}

```

**Figura 4.10:** Estructura de la entidad Device

Seguindo la estructura Fiware de la [Gestión de residuos urbanos de Badajoz](#), podríamos sensorizar tanto los contenedores de una ciudad como los camiones de recogida. Los sensores de los contenedores se utilizarían principalmente para medir el índice de llenado de un contenedor, su inclinación para posibles casos de caída, o medir aumentos de temperatura dentro del contenedor que puedan indicar que se ha prendido fuego. Por otro lado, los sensores de los camiones podrían indicarnos la posición de los mismos, su nivel de carga, etc. Esta información se almacena utilizando el modelo de datos de Device relatado anteriormente.

En definitiva, utilizamos la información proveniente de los distintos sensores (contenedores, cubos y camiones), y los almacenamos en los modelos de datos de Device. Posteriormente, en nuestra plataforma se mostrarán mapeados, embebidos en la web, que utilizarán los geojson generados por los sensores para ubicar en el mapa la localización de los distintos contenedores dentro de la ciudad. Además, junto con esta localización, se brindará información extra al usuario del contenedor proveniente también de los sensores como, por ejemplo, el nivel de llenado del contenedor o cubo seleccionado.

Por otro lado, se puede usar el modelo WasteManagement para almacenar toda la información proporcionada por las administraciones y empresas de forma privada, con la cuál podemos generar un histórico, almacenarlo en MySQL y mostrarlo posteriormente en PowerBi si se considera de interés para el usuario final.

Para generar este histórico de información, necesitamos utilizar un módulo de Fiware extra, ya que el Context Broker no almacena información si no que sólo muestra la última a la que ha tenido acceso. Este módulo es Cygnus.



**Figura 4.11:** Funcionamiento habilitador Cygnus

El usuario realiza peticiones GET a Cygnus, que está conectado, en este caso, a MySQL, y está suscrito al Context Broker, pendiente de cualquier cambio que este le notifique. Recordemos que estos cambios se les conoce como **cambios de contexto** y vienen dados por los generadores de contexto, que pueden ser desde usuarios, a sensores, etc.

#### 4.2.4 QGIS

**QGIS [8]** es un software libre y de código abierto de Sistema de Información Geográfica. Entre todas sus principales características, destaca la capacidad para visualizar, gestionar y analizar datos, generando mapas. El estudio de esta herramienta es de especial interés pues mediante la conexión con Fiware permitiría mostrar la ubicación de los contenedores y camiones en un mapa.



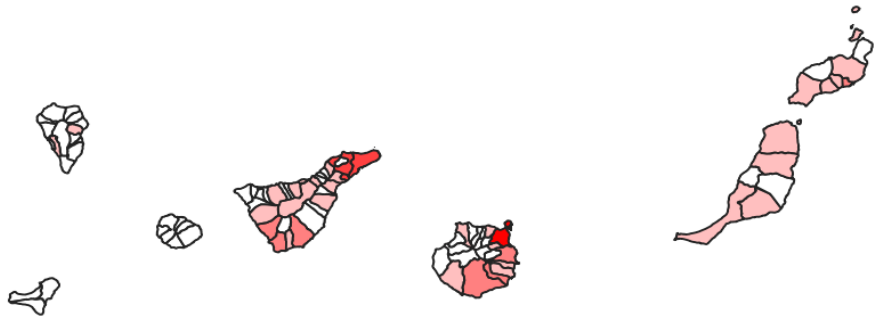
**Figura 4.12 :** Logo QGIS

Mucha de la información que se utiliza para el funcionamiento de la plataforma está georreferenciada. Este tipo de información capacita para la elaboración de mapas y realización de análisis. Por ejemplo, en la plataforma se utiliza dicha información georreferenciada para : localización de elementos del sistema de recogida de residuos, como contenedores, papeleras y camiones.

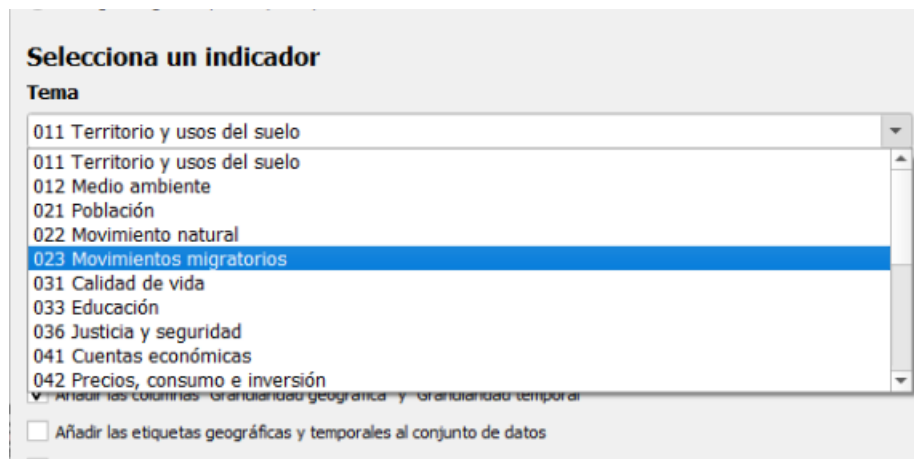
Como toma de contacto con esta nueva herramienta, se elabora un mapa insular separado por municipios de las islas y coloreado según distintos rangos de población.



Este coloreado va en función a distintos rangos de población. Este mapa se pudo realizar gracias a que el propio ISTAC tiene un plugin para el QGIS, donde se permite seleccionar distintas temáticas (Población, Educación..)



**Figura 4.13:** Mapeado de población con QGIS



**Figura 4.14:** Selector de tema en el plugin del ISTAC en QGIS

Después, dentro de cada temática se puede elegir un determinado indicador como valor de medición:

**Selecciona un indicador**

**Tema**

033 Educación

**Indicador**

Alumnado no universitario. Centros privados concertados. Enseñanzas de régimen general

**Formato de salida**

Visualizar las medidas como columnas

**Figura 4.15:** Selector de indicador en el plugin del ISTAC en QGIS(I)

Para el caso que nos ocupa en este Proyecto de Final de Grado, se selecciona como tema en el QGIS el Medioambiente, para así ver si se puede realizar alguna medición que ayude durante el desarrollo del proyecto. Sin embargo, este plugin oficial del ISTAC para el QGIS solo da como indicador de información dentro de medioambiente los accidentes medioambientales, lo cuál no es de gran utilidad en este caso.

**Selecciona un indicador**

**Tema**

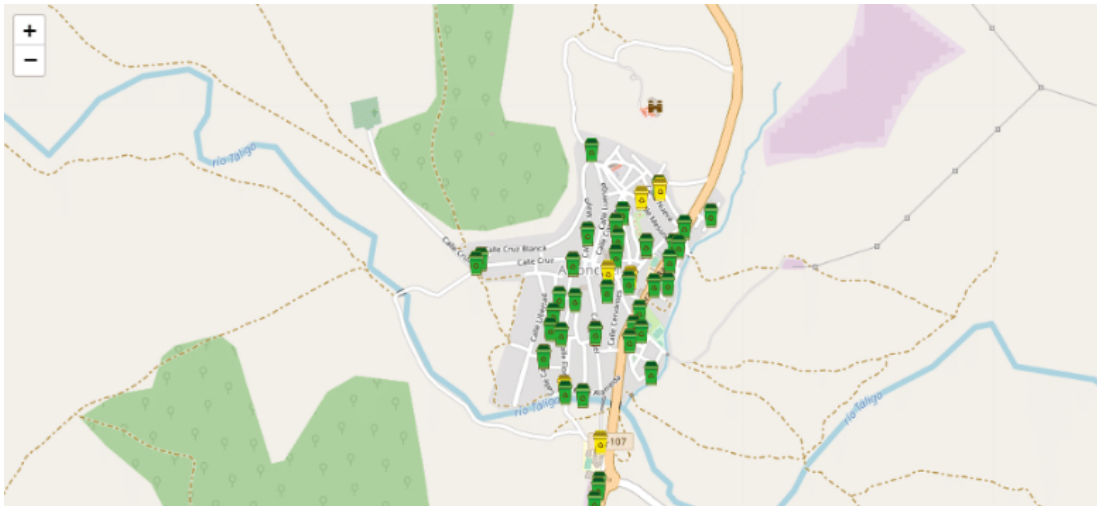
012 Medio ambiente

**Indicador**

Infracciones ambientales. Expedientes abiertos por la ACPMN

**Figura 4.16:** Selector de indicador en el plugin del ISTAC en QGIS(II)

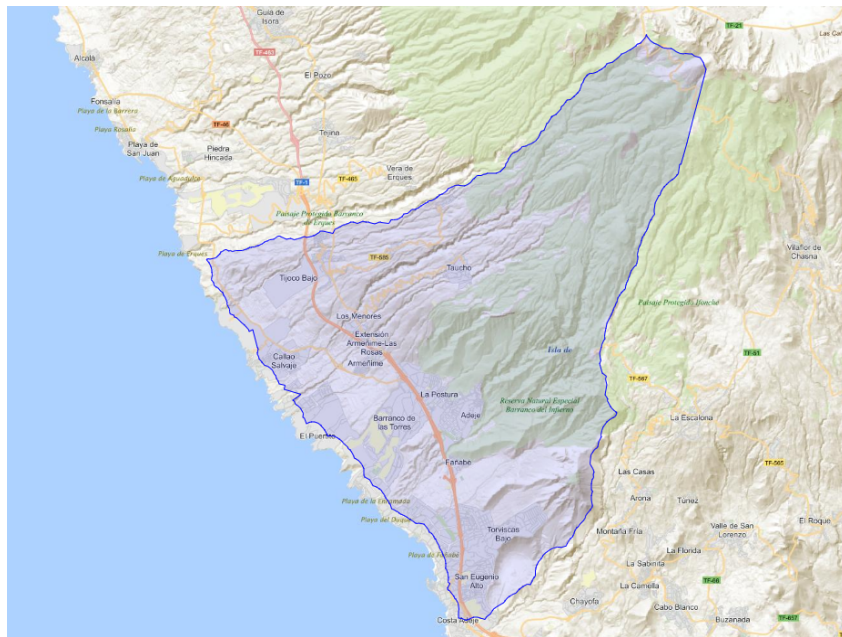
Sin embargo, con la información adecuada proporcionada por una determinada ciudad, se puede implementar en QGIS un sistema de mapeado donde se localizan, por ejemplo, todos los contenedores de una determinada urbe. Esto ayudaría tanto a los habitantes como a los trabajadores a conocer la ubicación exacta de los mismos y unido a la información almacenada por Fiware, también su estado actual(nivel de llenado, carga máxima,etc).



**Figura 4.17:** Mapa de Gestión de Residuos de Badajoz

Si una determinada ciudad, supongamos Adeje, proporciona información relacionada con los contenedores, como su geolocalización, podríamos crear mapas separados por capas, donde mostrar la ubicación de contenedores, papeleras, etc. A su vez, si se realizara una sensorización de los camiones, también se podría crear una capa en QGIS con la ubicación en tiempo real de los mismos. La idea principal es tener una serie de mapas de la ciudad georreferenciados que funcionen como visores de información relacionada con la gestión de residuos, como ocurre en el proyecto de gestión de residuos de [Badajoz](#). Basándonos en esto, se podría generar un mapa de Adeje separado por capas donde existan:

- Una capa destinada los contenedores, divididos por colores en función al tipo de residuo que contienen
- Una capa destinada a los cubos pequeños de basura que están por toda la ciudad
- Una capa donde están ubicados los camiones de basura en tiempo real.



**Figura 4.18:** Mapeado extensión de Adeje

Toda esta información vendría de la sensorización Fiware, que al utilizar un modelo de datos de como el Smart Sensing, nos podría dar la geolocalización de todo lo sensorizado (cubos, contenedores y camiones), mediante .geojson, permitiéndonos mostrarlo posteriormente en un GIS (QGIS en este caso).




**Figura 4.19:** Mapa de Adeje





La idea principal es que existan diversos mapas (capas en QGIS), que se muestren en la página web en función de si deseamos ver cubos, contenedores o camiones. La iconografía, para diferenciar las distintas capas en función a lo que se desea visualizar, sería la siguiente:

	<p>Imagen que hace referencia los cubos de basura pequeños que hay repartidos por el núcleo urbano.</p>
	<p>Imagen que hace referencia al contenedor de basura destinado a la retirada de vidrio.</p>
	<p>Imagen que hace referencia al contenedor de basura destinado a la retirada de envases de papel y cartón.</p>
	<p>Imagen que hace referencia al contenedor de basura destinado a la retirada de envases de plástico, latas y briks.</p>
	<p>Imagen que hace referencia al contenedor de basura destinado a la retirada de residuos orgánicos.</p>
	<p>Imagen que hace referencia al contenedor de basura destinado a la retirada de restos.</p>

	<p>Imagen que hace referencia al camión de recogida de residuos.</p>
---	--

**Tabla 4.4:** Iconografía QGIS

Unida a esta iconografía, se permitirá la visualización de información relacionada con el contenedor o cubo indicado (gracias a la sensorización con Fiware) como, por ejemplo, el nivel de llenado del mismo. Esto se podrá realizar gracias a los .geojson que se proporcionan directamente desde la plataforma Fiware, permitiendo a QGIS pintar los iconos en el mapa en función de las coordenadas (altitud y latitud) proporcionadas. QGIS brinda la posibilidad de, de manera sencilla, utilizar los .geojson como fuente de de datos vectoriales, utilizando las coordenadas que proporciona para ubicar en la capa que se desee de un mapa la localización de lo que se quiere mostrar.

Una vez creados estos mapas por capas, los podemos embeber en la web mediante QGIS 2 Web, un plugin de QGIS que nos permite generar iframes e integrarlos en nuestro portal web.

## 4.2.5 PowerBi

Para la elaboración de los cuadros de mando, se utiliza **PowerBi [19]**, una herramienta de visualización de datos de Microsoft que permite elaborar gráficas con los datos que se le proporcionen. Se ha escogido esta herramienta, sobre los habilitadores gráficos de Fiware, porque es más potente y permite realizar cuadros de mando con gráficas más visuales y fáciles de usar para el usuario final. PowerBi te permite seleccionar BigQuery y MySQL como fuente de datos externa, por lo que es igual de sencillo y directo la utilización de los datos si estos están localizados tanto en GCP (Google Cloud Platform) como en la plataforma Fiware.





**Figura 4.23:** Logo de PowerBi

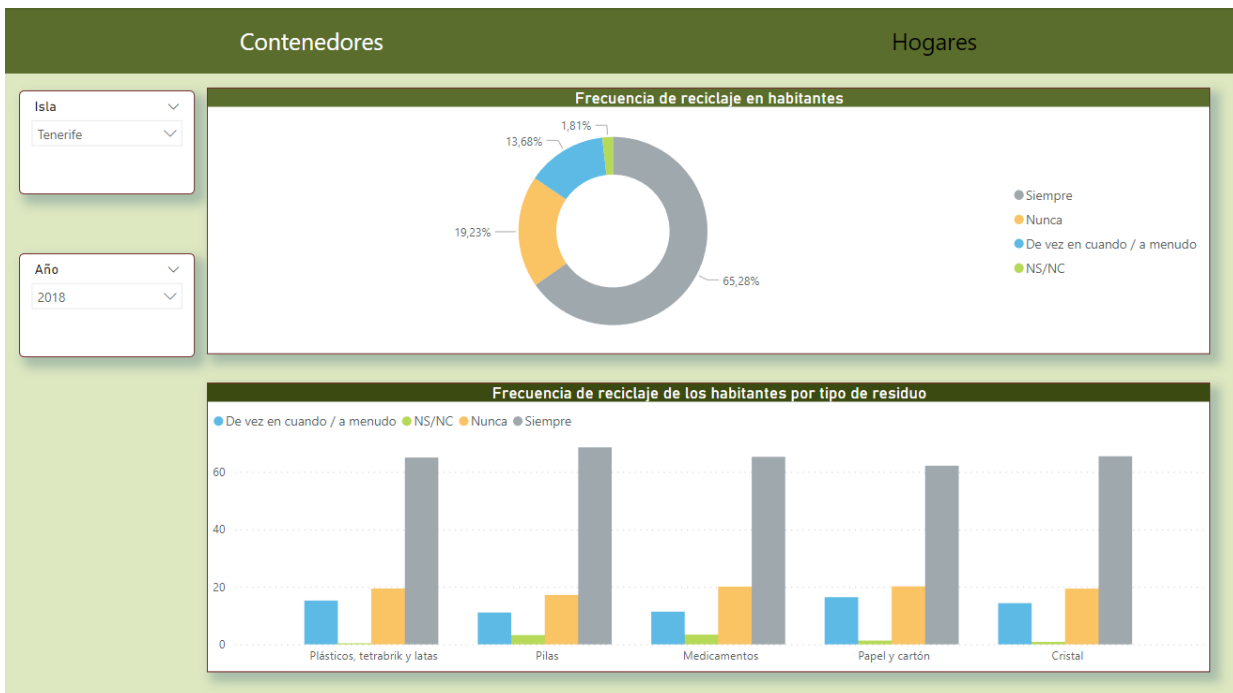
La idea principal es que los cuadros de mandos elaborados con PowerBi queden embebidos en el portal web final, de manera que toda información extraída, tanto de fuentes públicas y privadas como de los sensores, se muestre de una manera más sencilla y visual de cara al usuario.

En estas gráficas se pueden observar, separados por municipios de la isla, distintos campos relacionados con la retirada de residuos, como por ejemplo el número de contenedores existentes por habitante, la cantidad de residuos recogidos divididos por tipo, etc. Además, toda esta información estará separada en diversas **páginas** dentro del mismo cuadro de mando para mantener orden y cohesión. Cada una de estas páginas posee diversos **filtros** que pueden ser utilizados por el usuario en función a la ubicación y año de los que desea observar los datos.

Además, Powerbi permite generar **iframes** de los cuadros de mando para así poder publicarlos posteriormente en la web de la plataforma de manera sencilla. Los iframes son elementos **HTML** que pueden ser incrustados dentro otro **HTML** principal.



**Figura 4.24:** Cuadro de mando sobre contenedores y retirada de vidrio realizado en PowerBi



**Figura 4.25:** Cuadro de mando de frecuencia de reciclaje en habitantes

## 4.2.6 Wordpress

**Wordpress** [21] es un sistema de gestión de contenidos (CMS) que permite crear de manera intuitiva y sencilla portales web .



**Figura 4.26:** Logo de Wordpress

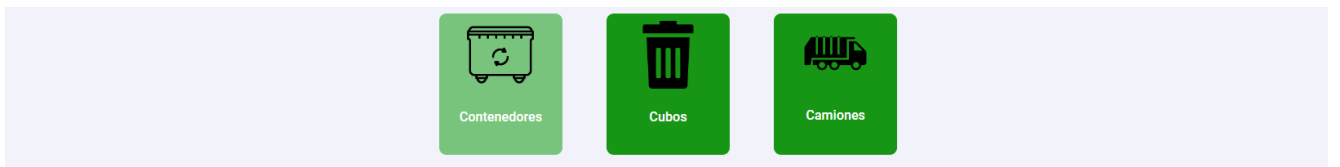
La página web donde estará alojada nuestra plataforma está realizada con esta herramienta y se estructura de la siguiente manera:

- Un header donde se vea el nombre de la plataforma.
- Una sección con botones donde poder variar entre las distintas categorías de cuadros de mando existentes.
- Un espacio destinado al cuadro de mando que se va a visualizar
- Un mapeado del municipio con la localización de los contenedores, cubos o camiones para los usuarios.

Se busca crear un entorno amigable, donde prime la usabilidad, y en el que los usuarios puedan acceder a la información elaborada, para que ésta les ayude a la hora de la toma de decisiones relacionadas con la gestión de residuos.



Figura 4.27: Mockup página web (I)



Gestión de residuos **Contenedores**

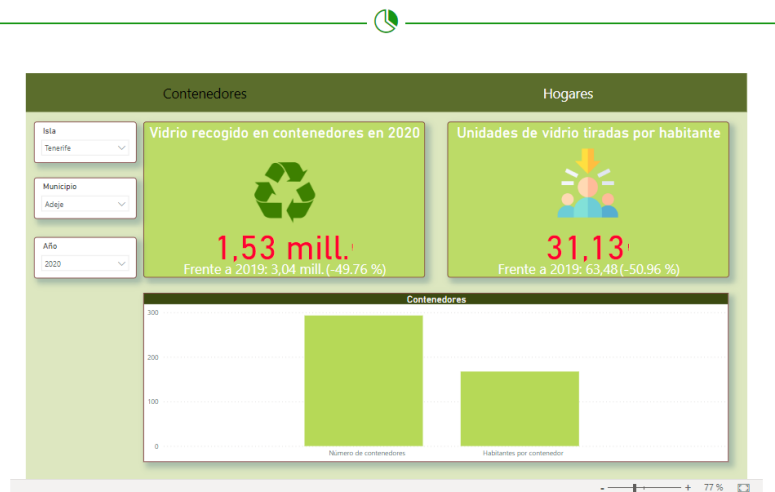


Figura 4.28: Mockup página web (II)



**Figura 4.29:** Mockup página web (III)

# Capítulo 5 : Conclusiones y líneas futuras

En este capítulo se describen las conclusiones alcanzadas tras la terminación del proyecto, así como las líneas futuras de mejora y desarrollo de la plataforma de gestión de residuos.

## 5.1 Conclusiones

El uso de las tecnologías Fiware se encuentra en crecimiento constante. Cada vez son más las ciudades europeas, incluidas las españolas, que optan por estas tecnologías para el desarrollo de estrategias de conversión en ciudades inteligentes. Los datos son uno de los recursos que más valor competitivo aportan a las organizaciones. Fiware facilita con los modelos de datos que se comparten en su comunidad el desarrollo de plataformas inteligentes.

El desarrollo de este tipo de plataformas no es sencillo, ya que entre otros aspectos, depende de la cantidad de información de la que se disponga. Por ello, es de vital importancia que las administraciones y empresas compartan esta información de forma abierta para que se le pueda dar buen uso y crear un impacto positivo en la sociedad.

Se han cumplido con éxito los objetivos trazados en el proyecto, el diseño y prototipado de la plataforma. La plataforma orientada a la visualización de datos sobre la gestión de residuos, ayuda a tomar decisiones más eficientes y eficaces a las empresas y administraciones. Con el análisis y otros procedimientos soportados en la información y conocimiento podemos ayudar a ahorrar recursos y a mejorar el medioambiente.

## 5.2 Líneas futuras

Actualmente, la plataforma diseñada en este proyecto captura, almacena y transforma, información relevante de la gestión de residuos proporcionada a través de distintas fuentes para su visualización en forma de gráficas y mapas. En el futuro, podría utilizarse esta información para implementar procedimientos más complejos e inteligentes que ayuden en la planificación. Un ejemplo de estas soluciones serían la obtención en tiempo real de las rutas óptimas para los camiones en función a la ocupación de los contenedores, ahorrando entre otros, tiempo y combustible.

Además, este diseño de plataforma de gestión de residuos es trasladable a otros ámbitos como el turismo. Con toda la información abierta disponible referente al turismo en las islas, se puede desarrollar una plataforma similar que ayude a empresas y administraciones en la toma de decisiones en comercialización, promoción y planificación turística.

# Capítulo 6 : Summary and Conclusions

This chapter describes the conclusions reached after the completion of the project, as well as the future lines of improvement and development of the waste management platform.

## 6.1 Conclusions

The use of Fiware technologies is constantly growing. More and more European cities, including Spanish ones, are opting for these technologies for the development of smart city conversion strategies. Data is one of the resources that contributes the most competitive value to organizations. Fiware facilitates the development of intelligent platforms with the data models that are shared in its community.

The development of this type of platform is not easy, since among other aspects, it depends on the amount of information available. For this reason, it is of vital importance that administrations and companies share this information openly so that it can be put to good use and create a positive impact on society.

The objectives outlined in the project, the design and prototyping of the platform have been successfully met. The platform oriented at visualizing data on waste management helps companies and administrations make more efficient and effective decisions. With analysis and other procedures supported by information and knowledge, we can help save resources and improve the environment.

## 6.1 Future lines

Currently, the platform designed in this project captures, stores and transforms relevant waste management provided through different sources for visualization in the form of graphs and maps. In the future, this information could be used to implement more complex and intelligent procedures to help with planning. An example of these solutions would be the obtaining in real time of the optimal routes for the trucks based on the occupation of the containers, saving, among other things, time and fuel.

In addition, this waste management platform design can be transferred to other areas such as tourism. With all the open information available regarding tourism in the islands, a similar platform can be developed to help companies and administrations in decision-making in tourism marketing, promotion and planning.



# Capítulo 7 : Presupuesto

En este capítulo se realiza un presupuesto aproximado de los costes de realización de todo este proyecto. Para realizar el cálculo del presupuesto total, se ha realizado una división en función de si se hace referencia al gasto destinado al software como servicio o al gasto correspondiente al coste del personal.

## 7.1 Software como servicio

Tecnología	Presupuesto
Google Cloud Platform	El presupuesto ha sido realizado con la calculadora de precios de los servicios de GCP, se puede visualizar el desglose en este <a href="#">enlace</a> <b>(41,30 euros/mes)</b>
Wordpress	<b>40 euros</b> por un dominio
Equipamiento informático	<b>1000 €</b> por equipo

**Tabla 7.1:** Presupuesto tecnologías

Para la parte tecnológica, hacemos uso de Google Cloud Platform con sus servicios de BigQuery ,Cloud Storage y Compute Engine, lo cual supondría un gasto de **41,30/mes**. Suponiendo que mantengamos la plataforma durante un año, esto supondría un gasto total de **495,6 euros**.

El coste de Wordpress corresponderá al precio del dominio en el que se va a alojar nuestra web, costando aproximadamente unos **40 € al año**.

El proyecto se desarrollará utilizando tres equipos informáticos, los cuales han costado aproximadamente unos **1000 €** por equipo.

El total del presupuesto tecnológico asciende a **3.535,60 €** por un año de mantenimiento de la plataforma.

## 7.2 Coste de personal estimado

Personal	Unidad de medida	Coste por unidad	Nº de unidades	Coste estimado
Programador Senior (Director del proyecto)	Euros por hora	60.00 €	240	14.400 €
Programador Junior	Euros por hora	25.00 €	240	6.000 €
Programador Junior	Euros por hora	25.00 €	240	6.000 €

**Tabla 7.2:** Presupuesto por coste de personal

El desarrollo total del proyecto, costaría un total de **26.600 €** de personal estimado. Este presupuesto se ha elaborado teniendo en cuenta la contratación de un programador senior y dos junior, con una inversión de 720 horas en el proyecto (240 horas por individuo), a lo largo de tres meses de desarrollo.

## 7.3 Presupuesto total

Mantenimiento de Tecnologías (un año)	3.535,60 €
Desarrollo de la plataforma	26.600 €
10% extra de imprevistos	2.813,56 €
<b>Total estimado</b>	<b>32.949,16 €</b>

**Tabla 7.3:** Presupuesto total

# Capítulo 8 : Algoritmos

## 8.1 Algoritmo : Extracción de datos abiertos

```
/******
```

```
*
```

```
*read_raw_medioambiente .py
```

```
*
```

```
*****
```

```
* AUTOR: Cristian González Alonso
```

```
*
```

```
* FECHA : 20/03/2022
```

```
*
```

```
* DESCRIPCIÓN: Script que extrae en crudo, en formato JSON, información pública relacionada con la gestión de residuos en canarias de Datos Abiertos Canarias y la carga en Cloud Storage.
```

```
*
```

```
*****/
```

```
import requests
import json
import os
from google.cloud import storage

"""
Raw de extracción de datos referentes a Residuos
"""

os.environ['GOOGLE_APPLICATION_CREDENTIALS'] = 'Path de las credenciales'

def raw_residuos():
    uuids = [
        "2a5d34a6-c97a-4c6c-b404-7b2e25061a92",
        "f49b8aea-fb28-4da4-926b-b7f58ac8cb23"
    ]

    cm = 'Superficies'
    for uuid in uuids:
        url =
f'http://www.gobiernodecanarias.org/istac/jaxi-istac/tabla.do?accion=jsonMtd
```

```

&uuidConsulta={uuid}'

    print(url)

    try:
        response = requests.request("GET", url)
        if response:
            exportToStorage(cm, json.dumps(response.json()), uuid)
        else:
            return "No data to show"
    except requests.exceptions.RequestException as e:
        raise SystemExit(e)

def exportToStorage(cm, data, tabla):
    """
    Exporta la respuesta RAW a Google Storage
    """
    client = storage.Client(project = "proyecto de GCP")
    bucket = client.get_bucket('TFG_bucket')

    blob = bucket.blob(f'Medioambiente_EconomiaCircular/{cm}/raw/{tabla}.json')

    try:
        blob.upload_from_string(data)
        print('Copied' + tabla + '.json to Google Storage
{0}'.format("Modified" if blob.exists() else "New"))
        return "Ok"
    except Exception as e:
        print(tabla + ".json does not exist {0}".format(e))
        return "Error"

if __name__ == "__main__":
    raw_residuos()

```

## 8.2 Algoritmo : Procesado y carga de los datos

```
/*
 *
 * etl_medioambiente .py
 *
 ****
 * AUTOR: Cristian González Alonso
 *
 * FECHA : 2/04/2022
 *
 * DESCRIPCIÓN: Script que utiliza la información en crudo guardada en Cloud Storage, la
 procesa y la almacena en dataframes de forma ordenada. Por último, almacena estos dataframes
 en Cloud Storage de nuevo para que puedan ser utilizados por las herramientas de visualización
 *
 *****/
```

```
import json
import pdb
import pandas as pd
import os
from datetime import datetime, timedelta
from google.cloud import storage

os.environ['GOOGLE_APPLICATION_CREDENTIALS'] = 'Path de las credenciales'

def ETL_residuos()->str:

    glass_recovered = ETL_residuos_01()
    homes_per_residues = ETL_residuos_02()
    glass_recovered['Dimension'] = 'Vidrio recogido'
    homes_per_residues['Dimension'] = 'Hogares y residuos'

    master_turists = pd.concat([
```

```

        glass_recovered,
        homes_per_residues

    ])
    print(master_turists)
    resp = exportCuratedToStorage(master_turists, 'Superficies')
    return resp

def convert_string_year_to_date(string_year_month:str)->datetime:
    year = string_year_month[0:4]
    month = string_year_month[4:6]
    return datetime(int(year), int(month), 1)

def search_period(period:str, periods:list)->str:
    for item in periods:
        if period in item:
            return item[period]

def search_origin_by_code(code:str, origins:dict)->str:
    if code in origins:
        return origins[code]

def search_concept_by_code(code:str, concepts:dict)->str:
    if code in concepts:
        return concepts[code]

def search_indicator_by_code(code:str, indicators:dict)->str:
    if code in indicators:
        indicator = ''
        if indicators[code] == "Superficie":
            indicator = "Superficie"
        elif indicators[code] == "Número":
            indicator = "Número"
        elif indicators[code] == "Perímetro municipal":

```

```

        indicator = "Perímetro municipal"
    elif indicators[code] == "Longitud de costa":
        indicator = "Longitud de costa"
    elif indicators[code] == "Distancia":
        indicator = "Distancia"
    elif indicators[code] == "Vidrio recogido de contenedores":
        indicator = "Vidrio recogido de contenedores"
    elif indicators[code] == "Vidrio recogido por habitante":
        indicator = "Vidrio recogido por habitante"
    elif indicators[code] == "Número de contenedores":
        indicator = "Número de contenedores"
    elif indicators[code] == "TOTAL":
        indicator = "TOTAL"
    elif indicators[code] == "Nunca":
        indicator = "Nunca"
    elif indicators[code] == "De vez en cuando / a menudo":
        indicator = "De vez en cuando / a menudo"
    elif indicators[code] == "Siempre":
        indicator = "Siempre"
    elif indicators[code] == "NS/NC":
        indicator = "NS/NC"
    elif indicators[code] == "Habitantes por contenedor":
        indicator = "Habitantes por contenedor"
    else:
        indicator = "Altitud"
    return indicator

def ETL_residuos_01()->str:
    tabla = "2a5d34a6-c97a-4c6c-b404-7b2e25061a92"
    cm = "Superficies"

    json_array = readRaw(cm, tabla)

    df =
pd.DataFrame(columns=['Isla', 'Municipio', 'Año', 'Indicador', 'Valor'])

```

```

        origin_codes = [origin_code for origin_code in
json_array['categories'][1]['codes']]
        origin_labels = [label for label in
json_array['categories'][1]['labels']]
        origins = dict(zip(origin_codes, origin_labels))

        indicator_codes = [indicator_code for indicator_code in
json_array['categories'][0]['codes']]
        indicator_labels = [label for label in
json_array['categories'][0]['labels']]
        indicators = dict(zip(indicator_codes, indicator_labels))

islands = {
    'Canarias': origin_labels.index('CANARIAS'),
    'Lanzarote': origin_labels.index(' LANZAROTE'),
    'Fuerteventura': origin_labels.index(' FUERTEVENTURA'),
    'Gran Canaria': origin_labels.index(' GRAN CANARIA'),
    'Tenerife': origin_labels.index(' TENERIFE'),
    'La Gomera': origin_labels.index(' LA GOMERA'),
    'La Palma': origin_labels.index(' LA PALMA'),
    'El Hierro': origin_labels.index(' EL HIERRO')
}

periods = []

for period_travel in json_array['categories'][2]['codes']:
    if len(period_travel) == 4:
        periods.append({period_travel : period_travel} )

for item in json_array['data']:

    data = item
    new_date = data['dimCodes'][2]
    period = search_period(new_date, periods)
    origin = search_origin_by_code(data['dimCodes'][1], origins)
    origin_index = origin_labels.index(origin)
        indicator = search_indicator_by_code(data['dimCodes'][0],

```



```

indicators)

    island = ''

    for isla in islands.items():
        if origin_index >= isla[1]:
            island = isla[0]
        else:
            break

    value = 0 if data['Valor'] == '..' or data['Valor'] == '.' else
data['Valor']
    df = df.append({
        'Municipio': origin,
        'Isla': island,
        'Indicador': indicator,
        'Año': period,
        'Valor': value
    }, ignore_index = True)

waste = df
print(df)
return waste

def ETL_residuos_02()->str:
    tabla = "f49b8aea-fb28-4da4-926b-b7f58ac8cb23"
    cm = "Superficies"

    json_array = readRaw(cm, tabla)

    df = pd.DataFrame(columns=['Isla', 'Frecuencia', 'Tipo
residuo', 'Año', 'Valor'])

    concept_codes = [concept_code for concept_code in
json_array['categories'][2]['codes']]
    concept_labels = [label for label in
json_array['categories'][2]['labels']]

```

```

concepts = dict(zip(concept_codes, concept_labels))

    origin_codes = [origin_code for origin_code in
json_array['categories'][0]['codes']]
    origin_labels = [label for label in
json_array['categories'][0]['labels']]
    origins = dict(zip(origin_codes, origin_labels))

    indicator_codes = [indicator_code for indicator_code in
json_array['categories'][1]['codes']]
    indicator_labels = [label for label in
json_array['categories'][1]['labels']]
    indicators = dict(zip(indicator_codes, indicator_labels))

periods = []

for period_travel in json_array['categories'][3]['codes']:
    if len(period_travel) == 4:
        periods.append({period_travel : period_travel} )

for item in json_array['data']:

    data = item
    new_date = data['dimCodes'][3]
    period = search_period(new_date, periods)
    origin = search_origin_by_code(data['dimCodes'][0], origins)
        indicator = search_indicator_by_code(data['dimCodes'][1],
indicators)
    concept = search_concept_by_code(data['dimCodes'][2], concepts)

    value = 0 if data['Valor'] == '..' or data['Valor'] == '.' else
data['Valor']
    df = df.append({
        'Isla': origin,
        'Frecuencia': indicator,
        'Tipo residuo': concept,
        'Año': period,

```

```

        'Valor': value
    }, ignore_index = True)

waste = df
print(df)
return waste

def read Raw(cm, tabla):
    client = storage.Client(project = "proyecto de GCP")
    bucket = client.get_bucket('TFG_bucket')

    blob =
bucket.blob(f'Medioambiente_EconomiaCircular/{cm}/raw/{tabla}.json')
    json_array = json.loads(blob.download_as_string())
    return json_array

def exportCuratedToStorage(data, tabla):
    client = storage.Client(project = "proyecto de GCP")
    bucket = client.get_bucket('TFG_bucket')

    blob =
bucket.blob(f'Medioambiente_EconomiaCircular/Superficies/curated/{tabla}.csv
')

    try:
        blob.upload_from_string(data.to_csv(index=False), 'text/csv')
        print('Copied' + tabla + '.csv to Google Storage
{0}'.format("Modified" if blob.exists() else "New"))
        return "Ok"
    except Exception as e:
        print(tabla + ".json does not exist {0}".format(e))
        return "Error"

if __name__ == "__main__":
    ETL_residuos()

```

# Bibliografía

- [1] AENOR. (2016). Ciudades inteligentes. Definición Atributos y requisitos. Madrid: AENOR.
- [2] BigQuery: almacén de datos en la nube | BigQuery: Almacén de datos en la nube | <https://cloud.google.com/bigquery?hl=es>
- [3] Google Cloud. <https://cloud.google.com/bigquery?hl=es> Cloud Storage |. (s. f.). Google Cloud. <https://cloud.google.com/storage?hl=es>
- [4] Compute Engine | Compute Engine: Máquinas virtuales (VM) |. (s. f.). Google Cloud. <https://cloud.google.com/compute?hl=es>
- [5] González Rosas, Leopoldo M.C. Inteligencia de negocios Business Intelligence (BI). En, M., Estratégica, P., Dirección, Y., Tecnología (base de datos iti-552)
- [6] Fiware - Open APIs for Open Minds. Fiware. <https://www.Fiware.org/> Documentación.
- [7] Hernández Trasobares, A. (2003). Los sistemas de información: evolución y desarrollo. Proyecto social: Revista de relaciones laborales, (10-11), 149-165. Retrieved from <http://dialnet.unirioja.es/servlet/oaiart?codigo=&#61;793097>
- [8] QGIS. <https://www.qgis.org/es/docs/index.html> España. (2022, 23 junio). Atos. <https://atos.net/es/spain>
- [9] Developers Catalogue. Componentes Fiware. <https://www.Fiware.org/catalogue/>
- [10] «Smart cities»: la revolución tecnológica llega a las ciudades. (s. f.-b). Iberdrola. <https://www.iberdrola.com/innovacion/smart-cities>
- [11] Smart Data Models. Fiware. <https://www.Fiware.org/smart-data-models/>
- [12] Google-cloud-storage. (2022, 8 junio). Librería Cloud Storage. <https://pypi.org/project/google-cloud-storage/>
- [13] Pandas - Python Data Analysis Library. (s. f.). Librería Pandas. <https://pandas.pydata.org/>
- [14] La plataforma Fiware - Aprende Fiware en Español. (s. f.). Fiware. [https://Fiware-training.readthedocs.io/es\\_MX/latest/ecosistemaFiware/plataformaFiware/](https://Fiware-training.readthedocs.io/es_MX/latest/ecosistemaFiware/plataformaFiware/)
- [15] LUNDQUIST, Eric. Data Quality Is First Step Toward Reliable Data Analysis. En Eweek, 5, 2013.

[16] Residuos Profesional - Reciclaje, gestión de residuos. (s. f.). Residuos Profesional.  
<https://www.residuosprofesional.com/>

[17] DTI I- Turístico, D. Guía para acelerar la REACTIVACIÓN de DESTINOS TURÍSTICOS INTELIGENTES (7 de enero de 2020)

<https://www.destinosinteligentes.es/formacion/guia-para-acelerar-la-reactivacion-en-el-contexto-del-covid19-destinos-turisticos-inteligentes/>

[18] Visual Studio Code - Code Editing. Redefined. (s. f.). Visual Studio Code.  
<https://code.visualstudio.com/>

[19] Visualización de datos | Microsoft Power BI. (s. f.). PowerBi.  
<https://powerbi.microsoft.com/es-es/>

[20] Wieder, B., & Ossimitz, M. (2015). The impact of business intelligence on the quality of decision making – A mediation model. *Procedia Computer Science*, 64, 1163-1171.  
doi:10.1016/j.procs.2015.08.599

[21] WordPress.com. (s. f.). WordPress.com: el alojamiento gestionado de WordPress más rápido y seguro. <https://wordpress.com/es/>