



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Medición y reducción de la huella de carbono sobre
plataformas web de adquisición de datos de empresas

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Morant Juan, Nicolas

Tutor/a: Molina Marco, Antonio

Cotutor/a externo: ORTIZ FUENTES, FAUSTINO

CURSO ACADÉMICO: 2022/2023

Resumen

Hoy en día es importante conocer la huella de carbono de una organización o empresa, ya que el precio de la luz va en aumento y debemos también preocuparnos por el medio ambiente. Esto nos sirve para poder controlar el consumo energético de nuestra empresa y las emisiones de gases de efecto invernadero, y buscar soluciones para intentar reducir esa huella de carbono.

Además, es una información cada vez más solicitada por algunos clientes que quieren consumir servicios de empresas que tienen sellos de calidad. La idea para el TFG es crear una plataforma web que sirva a las empresas para calcular su huella de carbono, con el fin de intentar reducir la huella de carbono, ahorrar en costes energéticos y contribuir a la lucha contra el cambio climático.

Consistiría en una web en la cual las empresas/organizaciones podrían darse de alta y en la cual podrían introducir los datos de consumo de todos sus dispositivos, después de esto se le devolvería un informe con los resultados del cálculo de la huella de carbono. Los datos se podrían introducir manualmente en la web o mediante un archivo tipo csv/json. También se almacenará un registro de las emisiones mensuales de la empresa y que puedan ver la evolución de su consumo de un mes a otro, accediendo a los diferentes informes de cada mes.

La comunicación se haría mediante una API REST que guardaría la información en una base de datos en la nube, también se podrían introducir los datos en la nube mediante un programa que hiciese llamadas a la API directamente. Las tecnologías que se usarán son Blazor en .NET para la página web, una API creada con APN.NET Core con Entity Framework y una base de datos SQL. El lenguaje principal del proyecto será C#.

Palabras clave:

contaminación, medioambiente, energía, web, huella de carbono

Abstract

Nowadays it's important to acknowledge the carbon footprint of an organization or company, not only because the price of electricity is continuously increasing or because we must also worry about the environment. This allows us to control our company's energy consumption and greenhouse gas emissions and find solutions to try to reduce that carbon footprint. It's also really important data that is being requested by some customers who want to consume services from companies that have quality seals.

The idea for the TFG is to create a web platform that serves companies to calculate their carbon footprint, to try to reduce their carbon footprint, save on energy costs and contribute to the fight against climate change.

It will be a website in which companies / organizations can register and in which they can enter the consumption data of all their devices, after this a report is returned with the results of the computation of the carbon footprint. The data can be input manually on the web or through a csv/json type file.

This will also store a record of the monthly emissions of the company so they can see the evolution of their consumption from one month to another, accessing the different reports for each month.

The communication will be done through a REST API that will save the information in a cloud database, the data can also be stored in the cloud through a program that makes calls to the API directly.

The technologies that will be used are Blazor in .NET for the website, an API created with APN.NET Core with Entity Framework and an SQL Database. The main language of the project will be C#.

Keywords:

contamination, environment, energy, web, carbon footprint

Índice general

Índice general	III
Índice de Figuras	V
1. Introducción	1
1.1 Motivación	3
1.2 Objetivos	3
1.3 Impacto esperado.....	4
1.4 Estructura.....	4
1.5 Convenciones	5
2. Estado del Arte	6
2.1 Crítica al estado del arte.....	7
2.2 Propuesta.....	8
3. Análisis del problema	9
3.1 Identificación y análisis de soluciones posibles	9
3.2 Solución propuesta.....	9
3.3 Alcance y especificación de requisitos	10
3.3.1 Alcance y lista de requisitos.....	10
3.3.2 Actores del sistema	11
3.3.3 Modelo base de datos	12
3.3.4 Casos de Uso	13
3.4 Plan de Trabajo	13
4. Diseño de la solución.....	17
4.1 Arquitectura del Sistema	17
4.2 Diseño Detallado	18
4.3 Tecnología utilizada	19
4.4 Mockups de la Interfaz	22
5. Desarrollo de la solución propuesta	24
5.1 Módulo de cálculo de la huella de carbono.....	24
5.2 Creación de la API REST	24
5.2.1 Autenticación en la API	24
5.2.2 Llamadas de la API.....	26
5.3 Creación de la base de datos	27
5.4 Creación de la página web	27
6. Implantación	31

6.1 Selección de proveedores de servicios online	31
6.2 Instalación	32
6.3 Pasos extra después de la implantación.....	34
7. Validación y pruebas.....	35
7.1 Pruebas funcionales de la API.....	35
7.2 Pruebas del sistema web	38
8. Conclusiones	39
8.1 Relación del trabajo desarrollado con los estudios cursados.....	39
9. Trabajos Futuros.....	41
10. Referencias	42
Anexo I: Objetivos de desarrollo sostenible	43
Anexo II: Descripción de Casos de Uso.....	45
Anexo III: Clases y atributos	49
Anexo IV: Código generación y Validación token JWT	54
Anexo V: Llamadas a la API.....	55
Glosario	67

Índice de Figuras

Figura 1.1 Gráfico de gases y su contribución al calentamiento global [2]	2
Figura 2.1 Tabla comparativa de características de los servicios de las diferentes empresas.....	7
Figura 3.1 Actores del Sistema	11
Figura 3.2 Modelado de la base de datos	12
Figura 3.3 Casos de uso del sistema Web.....	13
Figura 3.4 Diagrama de Gantt del proyecto	14
Figura 3.5 Tiempo estimado y tiempo dedicado	14
Figura 3.6 Listado de desglose de paquetes de trabajo	15
Figura 4.1 Arquitectura del Sistema	17
Figura 4.2 Funcionamiento de Blazor Server	21
Figura 4.3 Mockup de inicio de sesión en la web.....	22
Figura 4.4 Mockup de visualización de dispositivos	22
Figura 4.5 Mockup de visualización de informes.....	23
Figura 4.6 Mockup de interfaz de añadir nuevo dispositivo a una empresa	23
Figura 5.1 Ejemplo de header y payload de un JWT en json	25
Figura 5.2 Ejemplo de las URL de llamadas para la clase Usuario	26
Figura 5.3 Interfaz para dar de alta un dispositivo nuevo en el sistema	28
Figura 5.4 Alerta de error al fallar el inicio de sesión.....	29
Figura 5.5 Agregar un nuevo consumo a un dispositivo	29
Figura 5.6 Visualización de la lista de usuarios de la empresa (vista de administrador)	30
Figura 5.7 Visualización de la lista de vehículos de la empresa (vista de usuario corriente)	30
Figura 6.1 Archivo de configuración docker-compose.yml	33
Figura 7.1 Uso de Postman para realizar la petición de inicio de sesión a la API.....	35
Figura 7.2 Mensaje de error de la API después de una petición fallida	36
Figura 7.3 Resultado de una petición que devuelve la lista de vehículos de la empresa	36
Figura 7.4 Directorio de carpetas con todas las peticiones creadas para probar la API	37
Figura 11.1 Código de validación de un token JWT	54
Figura 11.2 Código de generación de un token JWT.....	54
Figura 11.3 URL de las llamadas sobre las clases Electricidad y su consumo	55
Figura 11.4 URL de las llamadas sobre las clases Instalación Fija y su consumo.....	56
Figura 11.5 URL de las llamadas sobre las clases Maquinaria y su consumo	57
Figura 11.6 URL de las llamadas sobre la Organización	58
Figura 11.7 URL de las llamadas sobre los consumos de otras fuentes.....	58
Figura 11.8 URL de las llamadas sobre las clases Transporte y su consumo	59
Figura 11.9 URL de las llamadas sobre las clases Vehículo y su consumo.....	60
Figura 11.10 URL de las llamadas sobre los usuarios.....	61
Figura 11.11 URL de las llamadas de información de valores permitidos	61
Figura 11.12 URL de las llamadas de informes, parte 1	62
Figura 11.13 URL de las llamadas de informes, parte 2	63
Figura 11.14 URL de las llamadas de informes, parte 3	64
Figura 11.15 URL de las llamadas sobre las clases EmisionesFugitivas y su consumo	65

1. Introducción

Hoy en día, el cambio climático es la mayor amenaza medioambiental a la que se enfrenta la humanidad. El cambio climático es la modificación de la temperatura y del resto de variables del clima, que se está produciendo con una velocidad e intensidad sin precedentes en la historia de la humanidad, como consecuencia de la actividad humana. Está ocurriendo en todo el mundo y sus consecuencias pueden ser devastadoras, tanto para el medio ambiente como para las personas. La actividad humana, en concreto y principalmente, la quema de combustibles fósiles que generan gases de efecto invernadero, es la causa detrás de esta grave amenaza medioambiental.

La temperatura media mundial ha aumentado $1,11 \pm 0,13$ °C desde la época preindustrial. Las emisiones de gases de efecto invernadero a nivel global siguen aumentando cada año.

Según el Grupo Intergubernamental de Expertos sobre el Cambio Climático (IPCC), el nivel del mar podría subir entre 26 y 82 cm y la temperatura podría aumentar desde los 0,3 °C hasta los 4,8 °C en el peor escenario posible para finales de este siglo [1]. El cambio climático es un problema global que alcanza una perspectiva ambiental, política, económica y social en la que las peores previsiones también implican enormes pérdidas económicas. Y es que cuanto más tarde se actúe, mucho más elevadas serán las inversiones para la adaptación al aumento de la temperatura y puede llegarse a un límite en que la adaptación ya no sea posible.

El principal causante del cambio climático es el efecto invernadero. El efecto invernadero es un proceso en el que la radiación térmica emitida por la superficie planetaria es absorbida por los gases de efecto invernadero (GEI) atmosféricos y es irradiada en todas las direcciones. Dado que parte de esta radiación es devuelta hacia la superficie terrestre y la atmósfera inferior, ello resulta en un incremento de la temperatura superficial media respecto a lo que habría en ausencia de los (GEI).

Muchos de esos gases de efecto invernadero se producen de forma natural, pero debido a las actividades humanas, las concentraciones de algunos de ellos están aumentando en la atmósfera, los principales gases de efecto invernadero se describen a continuación.

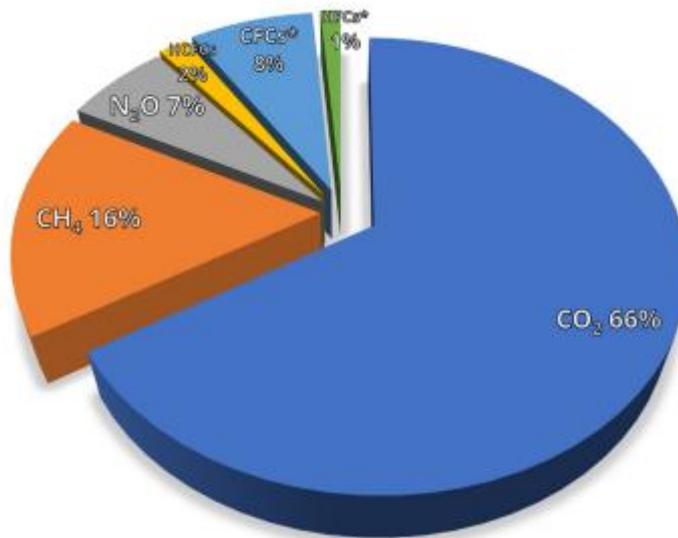


Figura 1.1 Gráfico de gases y su contribución al calentamiento global [2]

En la Figura 1.1 podemos ver los siguientes porcentajes de gases que contribuyen al calentamiento global:

El Dióxido de carbono (CO₂), que contribuye en un 66% al efecto del calentamiento del clima.

El Metano (CH₄), es el que causa un 16% del efecto de calentamiento provocado por los gases de efecto invernadero de larga vida, remarcar que un 40% del CH₄ que se emite en la atmósfera procede de fuentes naturales.

El Óxido Nitroso (N₂O) es el causante del 7% del efecto de calentamiento provocado por los gases de efecto invernadero de larga vida.

Los gases fluorados, son un conjunto de gases que, entre todos ellos, representan aproximadamente el 11% restante [2].

Un estándar para clasificar las emisiones de gases de efecto invernadero es por su origen, el más usado, incluso por la Unión Europea [3] es el GreenHouse Gas Protocol, que ha desarrollado un sistema que las divide en tres categorías, emisiones de Alcance 1, Alcance 2 y Alcance 3 [4].

Este sistema ayuda a medir el progreso que se realiza en cuanto a reducción de emisiones para por ejemplo, cumplir el objetivo del Acuerdo de París, en el cual se quiere limitar el aumento de la temperatura global este siglo a 1,5 grados Celsius reduciendo las emisiones de gases con efecto invernadero [5].

Las emisiones de Alcance 1, son aquellas que son directas, es decir, son las producidas por la propia empresa. Pueden ser resultado de ciertas acciones como la fabricación de productos, uso de vehículos, utilización de calefacción y ventilación.

Las emisiones de Alcance 2 son aquellas que se consideran indirectas y son producidas por la generación de electricidad, calor y vapor adquirido.

Las emisiones de Alcance 3 también son indirectas y engloban aquellas que son producidas por los clientes que consumen los productos de la empresa o por los proveedores que fabrican los productos utilizados por la empresa.

1.1 Motivación

Este trabajo trata sobre el cálculo de la huella de carbono de una organización o empresa. Esto es algo realmente importante si se tiene en cuenta aspectos como el medio ambiente, el uso de energía y el consumo responsable de esta. Cada vez más, la sociedad se está concienciando de la importancia de preocuparse por el planeta y cuidar el medio ambiente, aunque sea por una cuestión de necesidad.

Para contribuir a solucionar este problema, todos debemos de contribuir en la medida de lo posible. Una manera de hacerlo es que cada empresa controle sus emisiones de gases de efecto invernadero y su consumo de energía y que tome las medidas necesarias para reducir el impacto negativo de la empresa u organización en el medio ambiente. Además, hoy en día a los clientes les interesa cada vez más trabajar con empresas con un sello de calidad que trabajan de forma responsable por el medio ambiente.

Calcular la huella de carbono de una empresa puede ser tedioso, por lo que nuestro objetivo es crear una plataforma web que ayude a las empresas que lo necesiten a hacerlo de manera más sencilla y eficiente. En nuestro caso nos centraremos en calcular las emisiones de Alcance 1 y 2, ya que son aquellas sobre las cuales la empresa tiene más control y puede tomar medidas al respecto con mayor facilidad y rapidez.

Personalmente, me parece muy importante que todas las empresas intenten reducir su emisión de gases con efecto invernadero, para luchar contra el cambio climático y procurar que nuestro planeta siga siendo habitable en buenas condiciones para las generaciones futuras, no debemos pensar egoístamente y empezar a tomar medidas ahora mismo.

Este TFG se realiza en el ámbito de una práctica en la empresa Assessoria de Gestió i Comunicacions S.L (AGC), una empresa que se dedica al desarrollo de software de nuevos productos, resolución de incidencias y gestión y seguimiento de sus proyectos.

1.2 Objetivos

El principal objetivo de este trabajo es ofrecer una solución para el seguimiento de la huella de carbono de las empresas en España.

Como objetivos más específicos se encontrarían:

- Implementar un API web para el cálculo de la huella de carbono que permita a las empresas el registro de cualquier dispositivo que produzca gases de efecto invernadero y llevar un seguimiento de los datos de consumo.
- Implementar una alternativa de visualización de datos a través de la web más simple para las empresas que lo requieran.

1.3 Impacto esperado

El servicio ofrecido será muy útil para aquellas empresas que quieran saber su huella de carbono y tener un seguimiento de su consumo de energía , y a su vez, responsabilizar a las empresas para que reduzcan su impacto en el medio ambiente, ya sea consumiendo menos o implantando soluciones más sostenibles como el uso de energías renovables.

Este trabajo se relaciona con el ODS 12. Producción y consumo responsable y con el ODS 13. Acción por el clima porque se intenta contribuir a la lucha contra el cambio climático haciendo que las empresas sean conscientes de su impacto en el medio ambiente.

De esta manera, se espera que todos colaboremos para conseguir un futuro sostenible y minimizar el impacto del cambio climático, ofreciendo a las empresas una herramienta que les ayude a medir su huella de carbono y tomar las medidas necesarias.

1.4 Estructura

A continuación, se comentará la estructura de este TFG para indicar el contenido del que trata cada capítulo.

Capítulo 1: Introducción, en este capítulo encontraremos los primeros pasos que han llevado a la elección de la temática de este trabajo, explicando la motivación para ello, los objetivos que se quieren cumplir y el impacto que tendrá el proyecto sobre la sociedad actual.

Capítulo 2: Estado del Arte, aquí se explicará el contexto de la aplicación, buscando aplicaciones similares en el mercado y comparándolo con otros posibles trabajos similares. Es en este apartado donde se realizará la propuesta de la temática del trabajo a realizar.

Capítulo 3: Análisis del problema, el objetivo de este apartado es identificar y analizar los requisitos de la aplicación, elegir cuál será la solución para el problema planteado, y presentarla. Se definirá un plan de trabajo a seguir y se acordará el alcance del proyecto, y por último se realizará la especificación de requisitos del proyecto.

Capítulo 4: Diseño de la solución, una vez identificados los requisitos, se diseñará el sistema primero de manera más amplia, indicando los diferentes principales sistemas de los que estará compuesta la aplicación y después explicando más concretamente el diseño de las clases y sus relaciones, el diseño de base de datos y todo aquello que sea relevante. Por último, se comentarán y justificarán las tecnologías escogidas.

Capítulo 5: Desarrollo de la solución propuesta, en este apartado se describirá el desarrollo del proyecto, junto con los problemas y dificultades encontradas y las decisiones tomadas durante el desarrollo.

Capítulo 6: Implantación, en este capítulo se explicará el proceso seguido para la implantación de la solución.

Capítulo 7: Pruebas, en este capítulo se presentarán las pruebas realizadas para verificar el correcto funcionamiento de la aplicación desarrollada.

Capítulo 8: Conclusiones, una vez terminado el desarrollo, se realizará una conclusión sobre el trabajo realizado, y se aclarará si los objetivos definidos en el tercer

capítulo han sido cumplidos o no. También se relacionará el contenido del trabajo con los estudios cursados.

Capítulo 9: Trabajos futuros, aquí se comentará posibles ampliaciones del trabajo realizado y que flecos habrían estado bien acabar, pero no fue posible por falta de tiempo.

Capítulo 10: Referencias, capítulo que recogerá una lista de las fuentes de información que se han utilizado a lo largo de este proyecto.

Anexos, información adicional de interés que no se ha incluido en el cuerpo de la memoria. Glosario de términos para explicar palabras específicas del área en la que se ha trabajado. También contendrá un apartado describiendo la relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

En el anexo podremos encontrar toda la información adicional como los casos de usos detallados, algún trozo de código, los atributos de las clases importantes del proyecto, todas las llamadas a la API, los mockups de la interfaz y la relación que tiene este trabajo con las ODS.

Glosario, una vez finalizado el anexo, habrá un glosario que contendrá aquellas palabras específicas y acrónimos del área de la que trata es proyecto, se recomienda su lectura antes de empezar puesto que algunas palabras se repetirán con frecuencia.

Otro aspecto a destacar es que todas las imágenes son de elaboración propia excepto la Figura 1.1 que es sacada de un boletín como se indica en su referencia.

1.5 Convenciones

A lo largo del proyecto se empleará el texto en cursiva para mostrar código de comandos, que aparecerán los apartados **5. Desarrollo de la solución propuesta** y **6. Implantación**. La cursiva se usará también para referirse a términos anglosajones.

En el capítulo **2. Estado del Arte** se utilizará el subrayado para destacar el nombre de los TFG con los cuales se está comparando este proyecto. Además, se utilizará la negrita para destacar conceptos relevantes.

2. Estado del Arte

La situación actual de la tecnología en cuanto a las aplicaciones de cálculo de huella de carbono en una empresa ha avanzado significativamente en los últimos años. La creciente conciencia sobre el cambio climático y la importancia de reducir las emisiones de gases de efecto invernadero ha llevado a un aumento en el desarrollo y la adopción de herramientas tecnológicas para medir y gestionar la huella de carbono de las empresas.

Existen numerosas aplicaciones y plataformas en línea que permiten a las empresas calcular su huella de carbono. Estas herramientas recopilan datos sobre las actividades y operaciones de la empresa, como el consumo de energía o el transporte. Luego, utilizan los factores de emisión de los gases para calcular las emisiones de gases de efecto invernadero asociadas con estas actividades.

A continuación, se han recopilado unas cuantas aplicaciones que existen actualmente en el mercado, y que realizan funciones muy parecidas a las que se propone desarrollar este TFG.

ClimatePartner, ofrece un servicio para calcular las emisiones de una empresa con un software en la nube y ayudar a implementar soluciones para reducir su impacto.¹

ClimateSelectra cuenta con una calculadora online para particulares que no es muy rigurosa, puesto que es un cuestionario con preguntas y respuestas ambiguas. También tienes una consultora que te ayuda a calcular y reducir la huella de carbono en la empresa.²

Huelladecarbono.info, es una calculadora online que te permite calcular de manera sencilla pero no muy rigurosa, la huella de carbono de algunos componentes como vehículos y electricidad.³

Repsol, tiene también un software de medición y cálculo de huella de carbono, además guardará un histórico de los consumos de los clientes.⁴

ClimateTrade, esta página tiene una calculadora online que al igual que Huelladecarbono.info, es muy sencilla con pocas opciones para hacer los cálculos. Cabe destacar que no es su principal servicio.⁵

A continuación se mostrará una tabla comparativa de características de los servicios de las diferentes empresas:

¹ <https://www.climatepartner.com/en/services/measure-carbon-footprints>

² <https://climate.selectra.com/es/empresas/huella-carbono>

³ <https://www.huelladecarbono.info/calcular/>

⁴ <https://www.repsol.es/autonomos-y-empresas/soluciones-sostenibilidad/medicion-calculo-huella-carbono/>

⁵ <https://market.climatetrade.com/carbon-footprint-calculator/company>

Características	Desarrollo propio	ClimateSelectra	Huelladecarbono.info	Repsol	ClimateTrade	ClimatePartner
Cálculo de huella de carbono	✓	✓	✓	✓	✓	✓
Cálculo de huella de carbono exhaustiva	✓	✓	✗	✓	✗	✓
Medición de huella de carbono en cualquier momento	✓	✗	✓	✗	✓	✗
Servicio Online	✓	✓	✓	✓	✓	✓
Disponibilidad 24h/7	✓	✗	✓	✗	✓	✗
Generación de informes	✓	✓	✓	✓	✗	✓
Validez real para empresas en España	✓	✓	✗	✓	✗	✓
Almacenamiento de consumos en base de datos	✓	✓	✗	✓	✗	✓
Generación de informes en cualquier momento	✓	✗	✓	✗	✗	✗
Sugerencias para reducir huella de carbono	✗	✓	✗	✓	✗	✓

Figura 2.1 Tabla comparativa de características de los servicios de las diferentes empresas

Como se puede ver en la Figura 2.1, el principal problema es que los servicios que permiten calcular la huella de carbono en cualquier momento son calculadoras online poco exhaustivas que no ofrecen un resultado servible como **Huelladecarbono.info** y **ClimateTrade**. Aquellas empresas que permiten calcular la huella de carbono de forma exhaustiva no ofrecen un servicio en el cual se pueda calcular la huella de carbono en cualquier momento, por lo general estas empresas se dedican a calcular las emisiones en un periodo, y ofrecer medidas para reducir su impacto medio ambiental, como **ClimateSelectra**, **ClimatePartner** y **Repsol**.

El objetivo del desarrollo propio no es recomendar medidas para reducir ese impacto, sino que lo que se propone es un servicio que permita registrar los dispositivos de la empresa y sus consumos a medida que se producen, por lo que al estar guardados en una base de datos, el cliente podría ver su consumo en el periodo de tiempo deseado en cualquier momento, y saber al instante su huella de carbono.

2.1 Crítica al estado del arte

A continuación, nombraremos algunos TFG que están ampliamente relacionados con la temática de la huella de carbono. No se ha encontrado ningún proyecto cuyo título tuviese que ver con la creación de un software para calcular la huella de carbono en empresas.

Por un lado, tenemos algunos ejemplos de TFG que calculan la huella de carbono en alguna empresa en concreta, que son de utilidad escasa, puesto que ayudan solo a una empresa o sector:

-Cálculo de la huella de carbono en el engorde de la lubina. [6]

-Huella de carbono en una industria de bebidas. [7]

-Cálculo de la huella de carbono de una industria cervecera. [8]

Luego tenemos proyectos de análisis de la huella de carbono y de su cálculo, pero que simplemente sirven como información y no proponen una solución al problema:

-Análisis comparativo de modelos de reducción de emisiones de CO2 en el transporte marítimo. [9]

-Análisis energético y de emisiones de dióxido de carbono equivalentes del municipio de Sagunto entre los años 2010 y 2019. Definición y evaluación de las propuestas a implementar para el cumplimiento del Pacto de los Alcaldes para el Clima y la Energía. [10]

Se podría concluir este punto diciendo que el objetivo no es solamente calcular la huella de carbono para una empresa en concreto o analizar este tema, sino que se pretende proponer una solución a este problema ofreciendo un servicio de manera a contribuir con la lucha contra el cambio climático.

2.2 Propuesta

Este trabajo pretende crear un servicio para que las empresas puedan saber su impacto real de huella de carbono.

A diferencia de la mayoría de los trabajos sobre esta temática, el objetivo es crear una solución funcional y útil que se utilizará en entornos reales y que va más allá de un simple TFG. Este TFG se produce para la empresa anteriormente comentada, AGC, por lo cual esta aplicación será utilizada para ofrecer un servicio en línea.

3. Análisis del problema

3.1 Identificación y análisis de soluciones posibles

El objetivo de este proyecto es crear una plataforma que permita a las empresas medir su impacto de la huella de carbono, a partir de sus datos de consumo de sus dispositivos.

En este caso no había muchas soluciones posibles, puesto que este proyecto se desarrolla para una empresa, además se requería que fuese independiente, es decir, que pudiese ser utilizado por sí solo, sin necesidad de ninguna otra plataforma. Es por eso por lo que se decidió hacer una API, ya que facilita la integración a otras plataformas y su implementación solo depende de ella misma.

Para la visualización y gestión de datos de forma más sencilla se propusieron dos alternativas, una aplicación de escritorio o móvil, o una aplicación web. Para este caso se acabó decantando por la web app puesto que permite ser ejecutada en cualquier dispositivo con el navegador, y no habría que adaptar la app para cada sistema operativo.

3.2 Solución propuesta

La solución elegida para este proyecto es la creación de una API de cálculo de huella de carbono junto con una web para la visualización de los datos.

Este desarrollo pasará por las siguientes fases :

Lo primero será investigar funcionamiento de la calculadora de la huella de carbono, buscar los factores de emisión, aprender el funcionamiento de las fórmulas y los parámetros necesarios para ello.

Luego estará toda la fase de especificación de requisitos , diseño de la arquitectura del sistema, modelado del sistema, casos de uso, para que a la hora de empezar a desarrollar la aplicación esté todo claro. También se crearán los prototipos de interfaz, y el modelado de bases de datos.

Luego se creará la plataforma de cálculo de huella de carbono, con las diferentes tablas y todos los factores de emisión necesarios, que será integrada a la API.

A continuación, se creará la API, todas las llamadas de gestión de dispositivos, de consumos y de autenticación de usuarios por roles mediante un *middleware* que usará JWT.

Una vez esté creada la API, se creará una página web que se apoye en esta API para ofrecer una experiencia más atractiva a aquellos clientes menos experimentados, que tal vez no sepan o no quieran hacer llamadas directamente a la API.

Para la implantación de la API se usará un proveedor en línea para desplegar el servicio a ofrecer usando contenedores de Docker.

Para las pruebas se utilizará principalmente Postman para las pruebas funcionales de la API, y para la web se usarán usuarios que comprueben el funcionamiento de la interfaz.

La lista de tareas a realizar quedaría de la siguiente manera:

- Investigación preliminar y diseño del concepto
- Diseño de la Arquitectura del Sistema
- Análisis de riesgos y situación actual de mercado
- Modelado conceptual (Casos de Uso, Diagramas UML, Historias de usuario)
- Módulo/Librería para cálculo de huella de carbono en función del tipo de emisión (Alcance 1, Alcance 2)
- Modelado de base de datos y diagramas de entidad-relación
- Sistema de BACKEND con microservicios API REST
- Ingesta y almacenamiento de la información recibida de los dispositivos en base de datos
- Sistema web FRONTEND
- Validación y pruebas

3.3 Alcance y especificación de requisitos

3.3.1 Alcance y lista de requisitos

A la hora de planificar el proyecto se debe de establecer el alcance de este y especificar los requisitos de manera detallada, para que sea claro el trabajo y el contenido a realizar con antelación.

Este TFG tiene un tiempo de dedicación estimado de 360 horas, repartidos entre el desarrollo del proyecto y la escritura de la memoria, por lo que, a la hora de establecer el alcance del proyecto, se ha tenido en cuenta esto. La duración estimada del proyecto es de 270 horas, como veremos en el apartado 3.4 más adelante.

El listado de requisitos se divide en dos categorías, los requisitos funcionales (aquellos que describen lo que debe hacer el sistema), y los requisitos no funcionales (aquellos que describen cómo funciona el sistema).

Requisitos Funcionales:

- RF01: El sistema permitirá que el usuario inicie sesión.
- RF02: El sistema permitirá al usuario dar de alta un dispositivo.
- RF03: El sistema permitirá al usuario añadir un consumo nuevo a un dispositivo.
- RF04: El sistema permitirá al usuario visualizar la información de su cuenta.
- RF05: El sistema permitirá al usuario visualizar la lista de dispositivos dados de alta.
- RF06: El sistema permitirá al usuario visualizar la lista de los consumos producidos.
- RF07: El sistema permitirá al usuario con derechos de administración de una organización eliminar un dispositivo.
- RF08: El sistema permitirá al usuario con derechos de administración de una organización modificar un dispositivo.
- RF09: El sistema permitirá al usuario con derechos de administración de una organización eliminar un consumo.
- RF10: El sistema permitirá al usuario con derechos de administración de una organización modificar un consumo.
- RF11: El sistema permitirá al usuario con derechos de administración de una organización modificar un usuario.

- RF12: El sistema permitirá al usuario con derechos de administración de una organización ver la lista de usuarios registrados dentro de su organización.
- RF13: El sistema permitirá al usuario con derechos de administración de una organización eliminar un usuario de su organización.
- RF14: El sistema permitirá al usuario con derechos de administración de una organización crear un usuario dentro de su organización
- RF15: El sistema permitirá al usuario con derechos de administración de una organización modificar la organización.

Requisitos No Funcionales:

- RNF01: El sistema tendrá un sistema de autenticación.
- RNF02: El sistema web podrá ser ejecutado en los siguientes navegadores modernos (Mozilla Firefox, Microsoft Edge, Opera, Brave, Google Chrome).
- RNF03: La disponibilidad del servicio será permanente, es decir, 24 horas al día.
- RNF04: El sistema web debe tener tiempos de respuesta rápidos a la hora de mostrar las listas.
- RNF05: La API debe tener una tolerancia a fallos muy alta, es decir, capturando las excepciones y mostrando mensajes de errores claros.
- RNF06: El servicio web debe ser intuitivo y fácil de usar para los usuarios.
- RNF07: El servicio web debe ser capaz de hacer las llamadas a la API correctamente.
- RNF08: La API debe actuar de manera independiente, facilitando así su integración a otros servicios.

3.3.2 Actores del sistema



Figura 3.1 Actores del Sistema

Como podemos observar en la Figura 3.1, en el sistema se distinguirán cuatro actores:

- El usuario sin registrar, que únicamente podrá realizar acciones como iniciar sesión o registrar la empresa
- El usuario corriente, que podrá acceder a las acciones básicas de la plataforma de cálculo de huella de carbono
- El administrador de la organización, que tendrá acceso a todas las funciones del servicio ofrecido, y la gestión de los roles de los usuarios de su empresa
- El super admin, que se trata de un rol que pertenece a un desarrollador de esta aplicación y sirve para cuestiones de gestión y mantenimiento.

3.3.3 Modelo base de datos

A continuación tenemos el modelo de la base de datos:

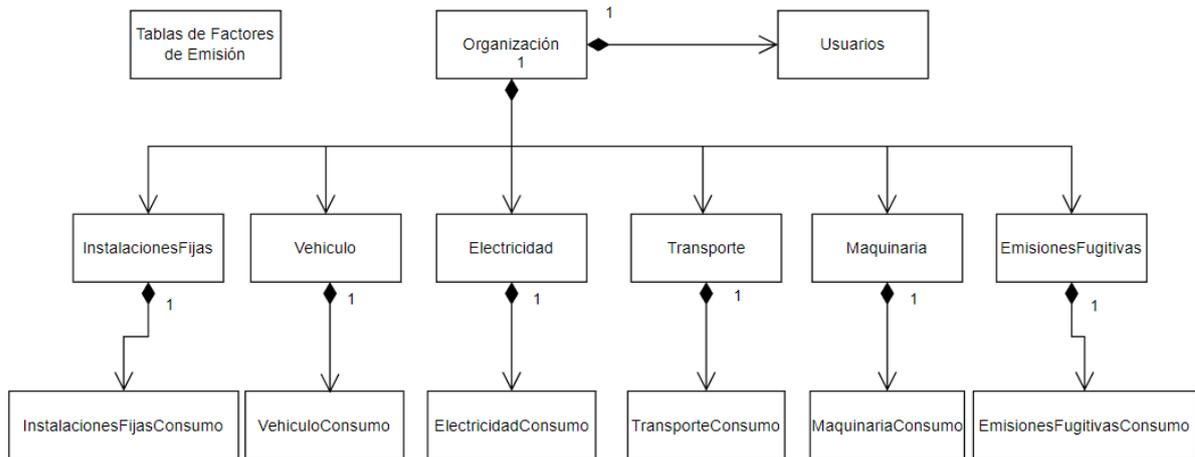


Figura 3.2 Modelado de la base de datos

Por una parte tenemos las tablas que contienen los factores de emisión, que no se relacionan con ninguna otra clase puesto que a estas tablas solo se accederá mediante consultas a la hora de realizar el cálculo de los consumos.

Por otra parte está la organización, que estará compuesta por diferentes usuarios y por diferentes tipos de dispositivos, por ejemplo, la tabla Electricidad hace referencia a cada dispositivo con consumo eléctrico que haya registrado la organización. Cada dispositivo tendrá asociados una serie de consumos, existen diferentes tablas para los consumos puesto que algunos dispositivos requieren atributos adicionales y además, con un gran volumen de consumos es más fácil de clasificar de esta manera y hará más eficientes las consultas SQL.

En el **Anexo III** se encontrarán las clases y sus atributos explicados con más detalle.

3.3.4 Casos de Uso

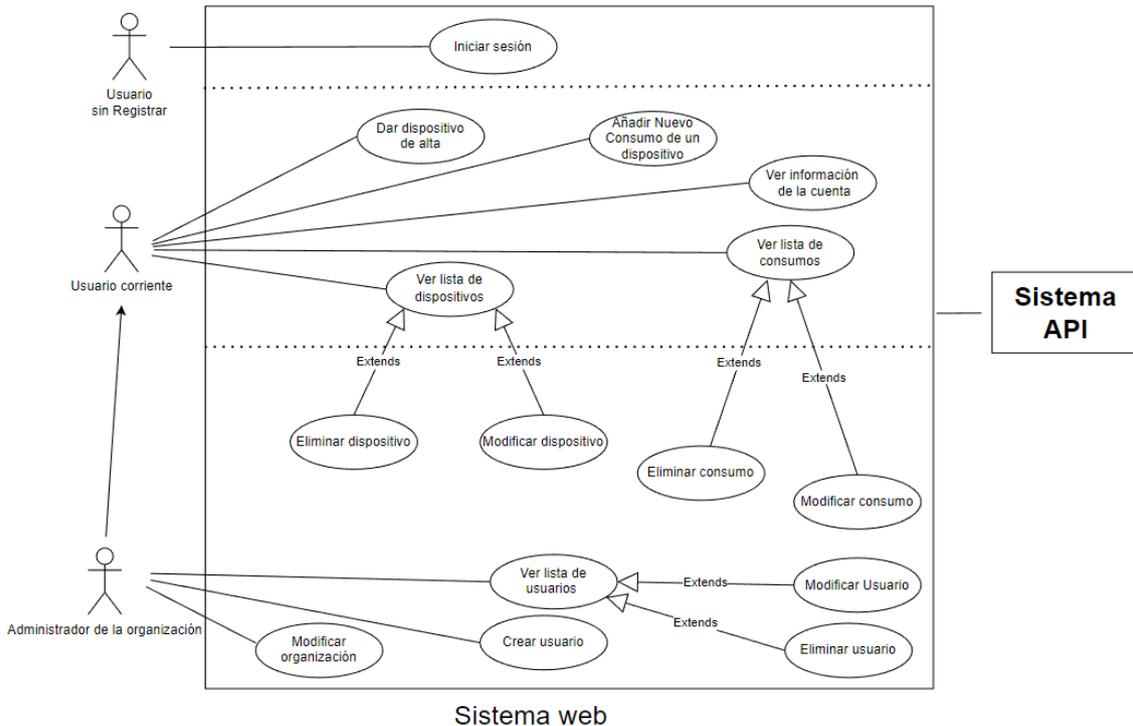


Figura 3.3 Casos de uso del sistema Web

Como se puede ver en la Figura 3.3, se muestra el modelado de casos de uso de la interacción del usuario con el sistema web, este interactúa con el sistema de la API.

Existen tres tipos de usuarios que interactúan en la web, y sus casos de uso están divididos por las líneas de puntos, se pueden observar varios casos de uso que extienden de otros como por ejemplo, el caso de uso Eliminar Dispositivo, que solo puede ser accedido por un usuario con el rol de administrador de la organización.

En el **Anexo II** se encuentran todas las descripciones detalladas de los casos de uso.

3.4 Plan de Trabajo

Para este proyecto usaremos las técnicas y herramientas de planificación de proyectos que se usan en la empresa, se usará Redmine, una herramienta para la gestión de proyectos, programada usando el framework Ruby on Rails y que además es de código abierto, lo que permite que sea modificada a gusto propio si se tiene conocimiento del lenguaje de programación Ruby.

Lo que se utilizará de esta herramienta serán los diagramas de Gantt, la creación y gestión de peticiones (tareas a realizar), y un plugin de tablero Kanban para una representación más vistosa del estado de las tareas. El diagrama de Gantt del proyecto es el mostrado en la Figura 3.4 a continuación:

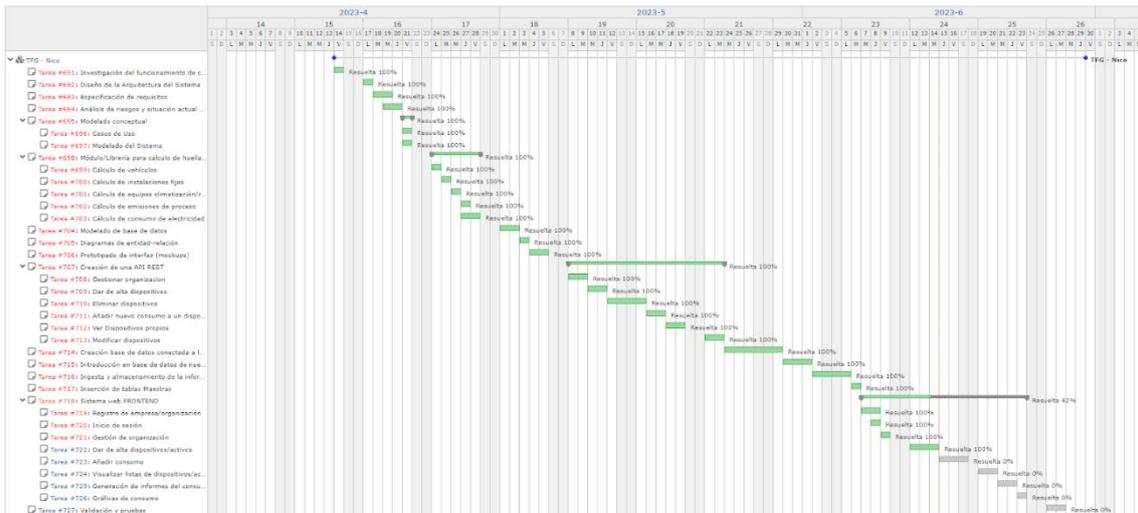


Figura 3.4 Diagrama de Gantt del proyecto

Para este desarrollo se ha decidido seguir una metodología tradicional mezclada con elementos característicos de la metodología ágil, las razones son que, al ser un proyecto con tareas a realizar muy claras y con un orden de prioridades preestablecido es más sencillo planificarlo de manera tradicional. Además, cada semana había una reunión corta para ver el estado del proyecto y ver el avance de este.

Por lo general, la estimación en horas y el trabajo realizado han sido bastante parecidos, con desviaciones menores, sin embargo, ha habido una fase que ha costado más horas de las previstas, la creación de la API, en concreto 16 horas más, por lo tanto, el tiempo estimado fue de 270 horas y el tiempo dedicado final unas 286 horas.

El desglose de paquetes de trabajo junto con la estimación en horas-persona para cada tarea, y la dedicación final se pueden observar en las Figuras 3.5 y 3.6:

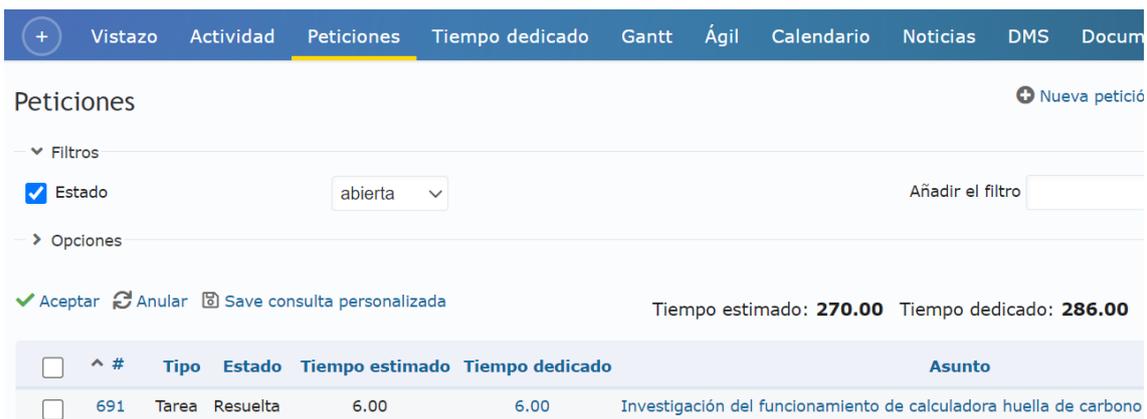


Figura 3.5 Tiempo estimado y tiempo dedicado

<input type="checkbox"/>	^ #	Tipo	Estado	Tiempo estimado	Tiempo dedicado	Asunto	
<input type="checkbox"/>	691	Tarea	Resuelta	6.00	6.00	Investigación del funcionamiento de calculadora huella de carbono	...
<input type="checkbox"/>	692	Tarea	Resuelta	5.00	6.00	Diseño de la Arquitectura del Sistema	...
<input type="checkbox"/>	693	Tarea	Resuelta	9.00	9.00	Especificación de requisitos	...
<input type="checkbox"/>	694	Tarea	Resuelta	7.00	7.00	Análisis de riesgos y situación actual de mercado	...
<input type="checkbox"/>	695	Tarea	Resuelta	0.00	0.00	Modelado conceptual	...
<input type="checkbox"/>	696	Tarea	Resuelta	3.00	3.00	> Casos de Uso	...
<input type="checkbox"/>	697	Tarea	Resuelta	3.00	3.00	> Modelado del Sistema	...
<input type="checkbox"/>	698	Tarea	Resuelta	0.00	0.00	Módulo/Librería para cálculo de huella de carbono	...
<input type="checkbox"/>	699	Tarea	Resuelta	5.00	5.00	> Cálculo de vehículos	...
<input type="checkbox"/>	700	Tarea	Resuelta	4.00	4.00	> Cálculo de instalaciones fijas	...
<input type="checkbox"/>	701	Tarea	Resuelta	4.00	4.00	> Cálculo de equipos climatización/refrigeración	...
<input type="checkbox"/>	702	Tarea	Resuelta	4.00	4.00	> Cálculo de emisiones de proceso	...
<input type="checkbox"/>	703	Tarea	Resuelta	6.00	9.00	> Cálculo de consumo de electricidad	...
<input type="checkbox"/>	704	Tarea	Resuelta	10.00	10.00	Modelado de base de datos	...
<input type="checkbox"/>	705	Tarea	Resuelta	5.00	5.00	Diagramas de entidad-relación	...
<input type="checkbox"/>	706	Tarea	Resuelta	8.00	10.00	Prototipado de interfaz (mockups)	...
<input type="checkbox"/>	707	Tarea	Resuelta	0.00	0.00	Creación de una API REST	...
<input type="checkbox"/>	708	Tarea	Resuelta	13.00	20.00	> Gestionar organización	...
<input type="checkbox"/>	709	Tarea	Resuelta	10.00	13.00	> Dar de alta dispositivos	...
<input type="checkbox"/>	710	Tarea	Resuelta	9.00	8.00	> Eliminar dispositivos	...
<input type="checkbox"/>	711	Tarea	Resuelta	11.00	8.00	> Añadir nuevo consumo a un dispositivo	...
<input type="checkbox"/>	712	Tarea	Resuelta	10.00	10.00	> Ver Dispositivos propios	...
<input type="checkbox"/>	713	Tarea	Resuelta	9.00	9.00	> Modificar dispositivos	...
<input type="checkbox"/>	714	Tarea	Resuelta	17.00	20.00	Creación base de datos conectada a la API REST	...
<input type="checkbox"/>	715	Tarea	Resuelta	10.00	15.00	Introducción en base de datos de nuevos dispositivos	...
<input type="checkbox"/>	716	Tarea	Resuelta	10.00	9.00	Ingesta y almacenamiento de la información recibida de los dispositivos en base de datos	...
<input type="checkbox"/>	717	Tarea	Resuelta	5.00	6.00	Inserción de tablas Maestras	...
<input type="checkbox"/>	718	Tarea	Resuelta	0.00	0.00	Sistema web FRONTEND	...
<input type="checkbox"/>	719	Tarea	Resuelta	7.00	7.00	> Registro de empresa/organización	...
<input type="checkbox"/>	720	Tarea	Resuelta	3.00	3.00	> Inicio de sesión	...
<input type="checkbox"/>	721	Tarea	Resuelta	5.00	6.00	> Gestión de organización	...
<input type="checkbox"/>	722	Tarea	Resuelta	15.00	10.00	> Dar de alta dispositivos/activos	...
<input type="checkbox"/>	723	Tarea	Resuelta	12.00	11.00	> Añadir consumo	...
<input type="checkbox"/>	724	Tarea	Resuelta	10.00	11.00	> Visualizar listas de dispositivos/activos	...
<input type="checkbox"/>	725	Tarea	Resuelta	10.00	11.00	> Generación de informes del consumo	...
<input type="checkbox"/>	726	Tarea	Resuelta	8.00	10.00	> Gráficas de consumo	...
<input type="checkbox"/>	727	Tarea	Resuelta	8.00	7.00	Validación y pruebas	...

Figura 3.6 Listado de desglose de paquetes de trabajo

Esto principalmente se debe a que no había experiencia en la creación de una API, además mientras se estaba implementando la API se decidió añadir la función de autorización y autenticación para poder restringir el acceso a ella y además poder definir diferentes roles a los usuarios con sus permisos adecuados, lo que retrasó aún más esta fase.

Durante la fase de creación de la página web, no se desvió demasiado en horas de trabajo con relación a las horas estimadas, porque ya se tenía experiencia con el desarrollo Front-end, pero a la hora de la creación de informes, se observó que hacían nuevas llamadas de la API a la base de datos para la extracción de ciertos datos en concreto, por lo que, de nuevo, se tuvo que invertir más horas en expandir la API.

La conclusión que podría sacarse de esta experiencia es que a la hora de estimar tareas sobre una tecnología desconocida o un tipo de tareas con el que no se tenga mucha experiencia se debería de tener en cuenta y añadir más horas de estimación. También se debe estimar pensando en las demás fases y repasar el desglose de paquetes por si faltara algo, por ejemplo, en este caso no se habían tenido en cuenta unas peticiones a la API que hacían falta para el apartado de informes y esto ha resultado en horas extra en una fase anterior del proyecto que en principio ya estaba terminada. Además, se denota una falta de experiencia a la hora de estimar horas puesto que es una tarea que se ha realizado pocas veces ya que la cantidad de proyectos en los que se ha participado es aún baja, cuatro en concreto.

4. Diseño de la solución

4.1 Arquitectura del Sistema

El proyecto desarrollado se compone de dos partes, por un lado, tenemos la API web que estará conectada a una base de datos MySQL y por el otro lado tendremos una página web que funcionará haciendo llamadas a dicha API.

La API web funciona de manera totalmente independiente a la página web, es decir, aquellas empresas que lo deseen podrían contratar únicamente el servicio de la API y gestionar ellos la información como deseen, ya sea creando sus propias páginas webs que usen la API o, por ejemplo, teniendo dispositivos eléctricos que envíen su información de consumo cada cierto tiempo de forma automática, y así controlar su consumo diario, semanal, mensual o anual.

La API web cuenta con un *middleware* de autenticación y autorización basado en JSON Web Token (JWT), lo que permite al sistema saber quién hace cada llamada y restringir si es el caso su acceso, esto permite que una empresa cliente pueda dar a sus empleados diferentes roles, lo que les permitirá acceder o no a ciertas funciones. Cabe destacar que este *middleware* está implementado dentro de la propia API, es decir, desde fuera la API y el *middleware* de autenticación y autorización son el mismo módulo.

Por otro lado, tenemos la página web que permitirá a los clientes utilizar este servicio de manera mucho más sencilla, dinámica y visual, mediante la web se harán las llamadas a la API que sacará información de la base de datos para mostrarla y poder tener una gestión más sencilla tanto de los dispositivos de la empresa, como de los usuarios de la empresa para los administradores de esta.

En la Figura 4.1 mostrada a continuación se ilustra como será la arquitectura del sistema explicada anteriormente.

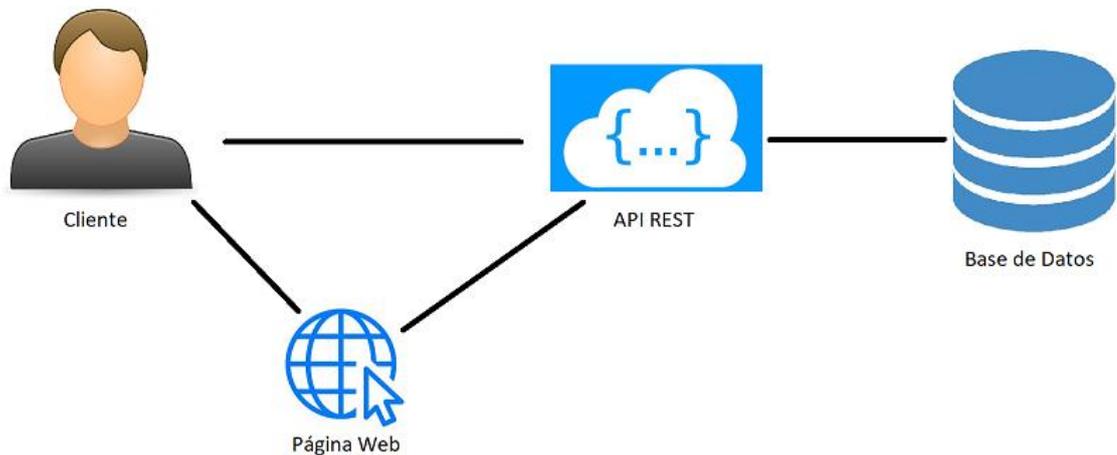


Figura 4.1 Arquitectura del Sistema

4.2 Diseño Detallado

Empezaremos detallando el módulo de la API web, describiendo los apartados por el que está compuesto :

- **Tablas Maestras:** Estructura de todas las tablas que servirán para calcular las emisiones, existirá una tabla con los factores de emisión correspondientes por cada tipo de dispositivo, es decir, seis en total.
- **Modelos:** Tendremos un modelo para la organización y un modelo para el usuario, después tendremos un modelo para cada tipo de dispositivo, un modelo para cada consumo de un tipo de dispositivo y por último un modelo para aquellos consumos que provengan de otras fuentes.

Los diferentes tipos de dispositivos que existen son:

- Vehículo
- Transporte
- Maquinaria
- Instalaciones Fijas
- Emisiones Fugitivas
- Dispositivos eléctricos

Cada modelo de consumo estará relacionado con el modelo del dispositivo asociado, es decir, el modelo **Consumo Vehículo** estará relacionado con el modelo **Vehículo**.

Por lo que en total tendríamos dieciocho tablas en la base de datos, contando las tablas maestras.

En el **Anexo III** se encontrarán las clases y sus atributos explicados con más detalle.

- **Autenticación:** *Middleware* para gestionar la autenticación y autorización de llamadas, habrá tres roles distintos, un Super admin para la gestión del servicio ofrecido, un administrador de organización, y un usuario corriente de la organización. Cada llamada de la API tendrá un nivel de autorización distinto dependiendo de si debe ser usada por uno u otro. Además, el *middleware* utilizado permite identificar cada usuario, entonces, al hacer una llamada a la API, se podrá saber a qué usuario pertenece.
- **Cálculos:** Estas clases serán llamadas cuando se añada el consumo de un dispositivo a la base de datos, teniendo en cuenta el tipo y sus propiedades, accederá a la tabla de factores de emisión correspondiente y calculará la emisión de ese dispositivo, agregándolo también a la base de datos.
- **Los Controladores:** Esta es la parte principal de este módulo, se compone de todas las llamadas que se podrán hacer a esta API, aquí se llamará a los métodos de las clases de cálculo para los consumos, se accederá al *middleware* para comprobar la sesión del usuario y se harán las consultas necesarias a la base de datos. Habrá una clase Controlador por cada clase Modelo, en concreto, una para cada dispositivo, una para cada consumo de un tipo de dispositivo, una para la organización y otra para el usuario.

A continuación, explicaremos los apartados del módulo web del sistema:

- **Servicios:** Se trata de aquellas clases que contiene los métodos necesarios para hacer las llamadas requeridas a la API. Existe una clase para cada controlador de la API, la función de los servicios es serializar los datos que introducirá el usuario en la web y encargarse de hacer la llamada a la API correspondiente. También deserializan los objetos recibidos de la API para poder mostrarlos de manera más atractiva. Los servicios recogen el token almacenado en el lado del cliente y se encargan de enviarlo en todas las cabeceras de las peticiones a la API, hasta que el usuario cierre sesión o el token expire.
- **Páginas Web:** Todas las diferentes páginas web que componen el servicio web ofrecido. Cada página web es un componente Razor que implementa HTML, CSS y lógica en C# que utilizará los métodos creados en los servicios para comunicarse con la API.
Habrá una página web para cada tipo y función de dispositivo, por ejemplo, una página para cada creación de dispositivo diferente, una para cada tipo de listas de dispositivos, y así con los informes, la modificación de dispositivos, la creación de consumos, la modificación de consumos, y las listas de consumos. Y también para la visualización del perfil, inicio de sesión, creación de usuario y modificación de usuario y organización.
Para la navegación existe una barra horizontal en la parte superior de cada página que es idéntica en todas ellas.

4.3 Tecnología utilizada

Para el desarrollo de este proyecto, se utilizará como Entorno de desarrollo integrado (IDE) Visual Studio 2022, esto nos permitirá tener los dos proyectos en un mismo lugar y facilitar el trabajo.

Por un lado, utilizaremos ASP.NET Core **Blazor**, un *framework* de Microsoft para crear interfaces de usuario web interactivas con .NET, y que además nos permitirá escribir código en C# en vez de Javascript.

Otra solución que se planteó fue hacer la web con Vue.js (un plugin de javascript), que es lo que se usa en la empresa, pero debido al poco conocimiento sobre Javascript se decidió hacerlo en Blazor, que es una mezcla de HTML y C#.

Por otro lado, crearemos una API web con ASP.NET Core en C# que se basa en Entity Framework Core, lo que nos facilitará la tarea de comunicarnos con la base de datos, esto se debe en parte a las consultas LINQ, que permiten en el código C# hacer consultas a la base de datos de forma sencilla y eficiente.

La base de datos se creará en MySQL por medio de las migraciones, esto significa que primero escribiremos el código de los modelos de clases y con ayuda del framework, ejecutaremos unos comandos que nos crearán las tablas en la base de datos automáticamente.

Por otra parte, las librerías que se han usado en este proyecto son Automapper ⁶, una librería para mapear objetos de un tipo a otro, Razor.SweetAlert ⁷ para la

⁶ <https://automapper.org/>

⁷ <https://github.com/Basaingeal/Razor.SweetAlert2>

implementación de las alertas en la web, y ChartJs.Blazor⁸ para el uso de las gráficas en el apartado de informes en la web.

Blazor

En Blazor, un componente es un elemento de la interfaz de usuario, los componentes son clases C# de .NET integradas en ensamblados de .NET, que informalmente se suelen llamar componentes Razor. Razor es una sintaxis para combinar marcado HTML con código en C# diseñada para aumentar la productividad del desarrollador, permite escribir en C# y en HTML en el mismo archivo.

En Blazor, los componentes se representan en una representación en memoria del elemento Document Object Model (DOM) del explorador denominado árbol de representación, que se usa para actualizar la interfaz de usuario de manera eficaz y flexible.

Blazor tiene dos modelos de hospedaje, Blazor Server y Blazor WebAssembly.

La principal diferencia entre estos dos es que en Blazor WebAssembly la aplicación se ejecuta en el cliente y no necesita una conexión persistente mientras que Blazor Server requiere una conexión constante, y provee comunicación en tiempo real entre cliente y servidor, lo que en nuestro caso es idóneo puesto que nuestra web app va a estar en constante comunicación con la API para hacer llamadas a la base de datos en tiempo real.

Blazor Server admite el hospedaje de componentes de Razor en el servidor de una aplicación de ASP.NET Core. Las actualizaciones de la interfaz de usuario se administran mediante una conexión de SignalR. El entorno de ejecución permanece en el servidor y controla la ejecución en código C# de la aplicación, el envío de eventos de interfaz de usuario desde el explorador al servidor y la aplicación de actualizaciones de una interfaz de usuario al componente representado que devuelve el servidor. [11]

⁸ <https://github.com/mariusmuntean/ChartJs.Blazor#samples>

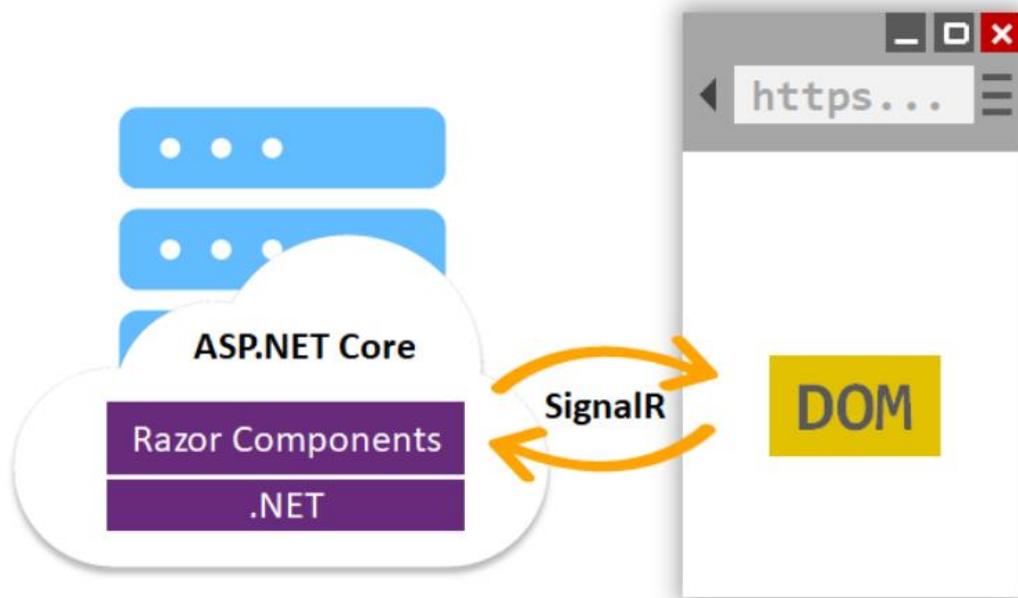


Figura 4.2 Funcionamiento de Blazor Server

Blazor Server genera un gráfico de componentes que se muestran de forma similar a un código HTML o XML Document Object Model (DOM). Este gráfico de componente incluye el estado del componente contenido en propiedades y campos.

Una vez realizada la conexión entre el cliente y el servidor, los elementos estáticos del componente representados previamente se reemplazan por elementos interactivos. La preprocesación del contenido en el servidor hace que la aplicación tenga más capacidad de respuesta en el cliente. Una vez que los componentes son interactivos en el cliente, la interacción del usuario y los eventos de la aplicación desencadenan las actualizaciones de la interfaz de usuario. [11]

4.4 Mockups de la Interfaz

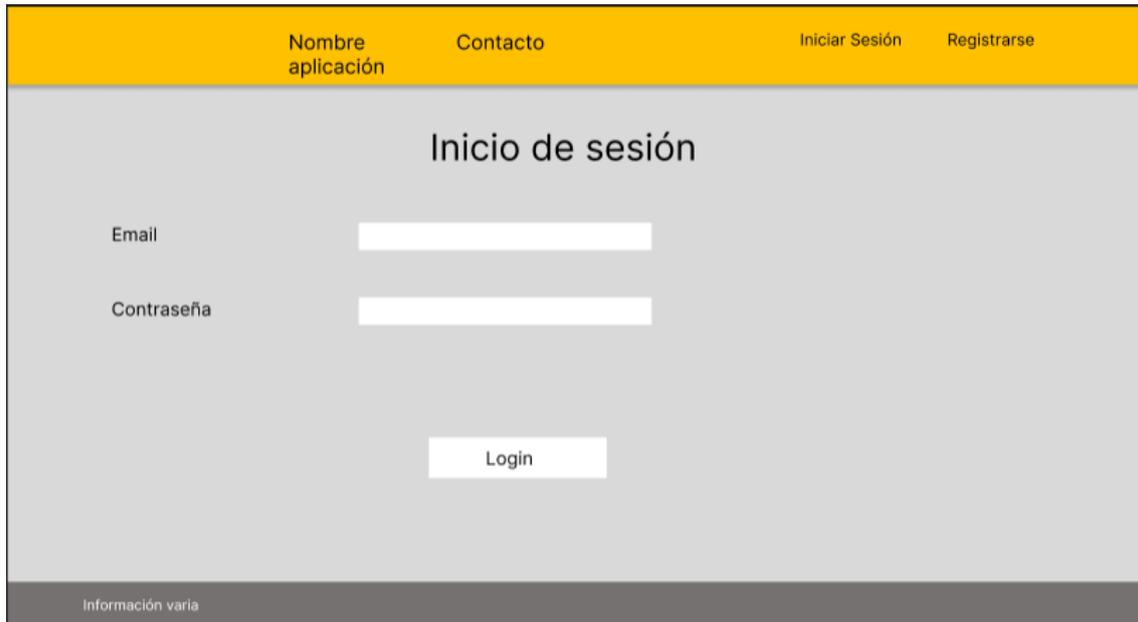


Figura 4.3 Mockup de inicio de sesión en la web

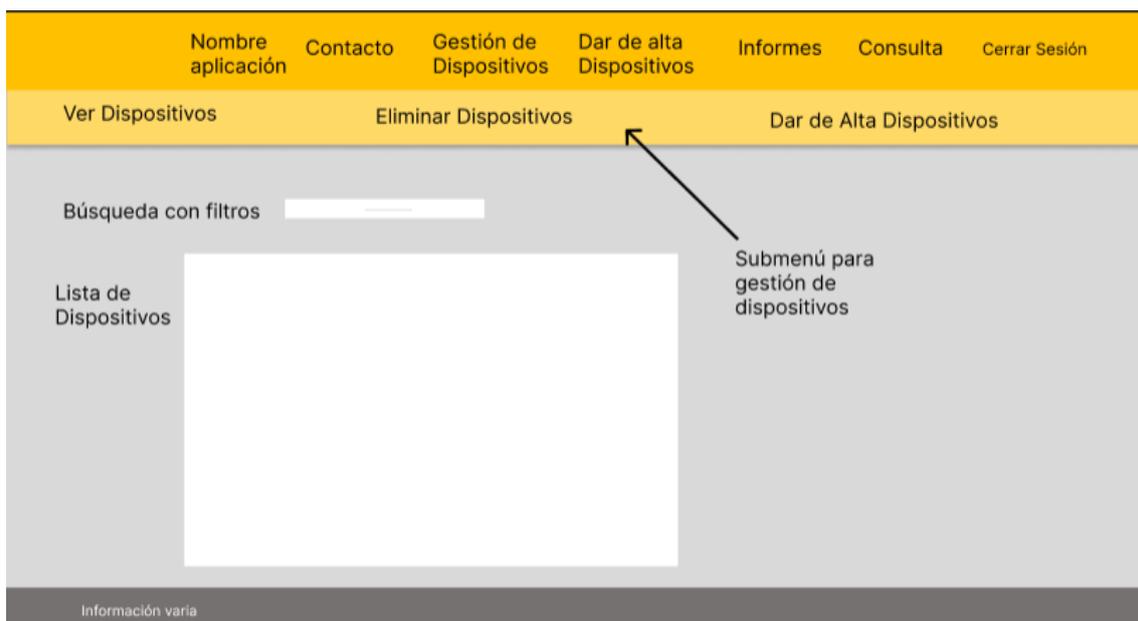


Figura 4.4 Mockup de visualización de dispositivos

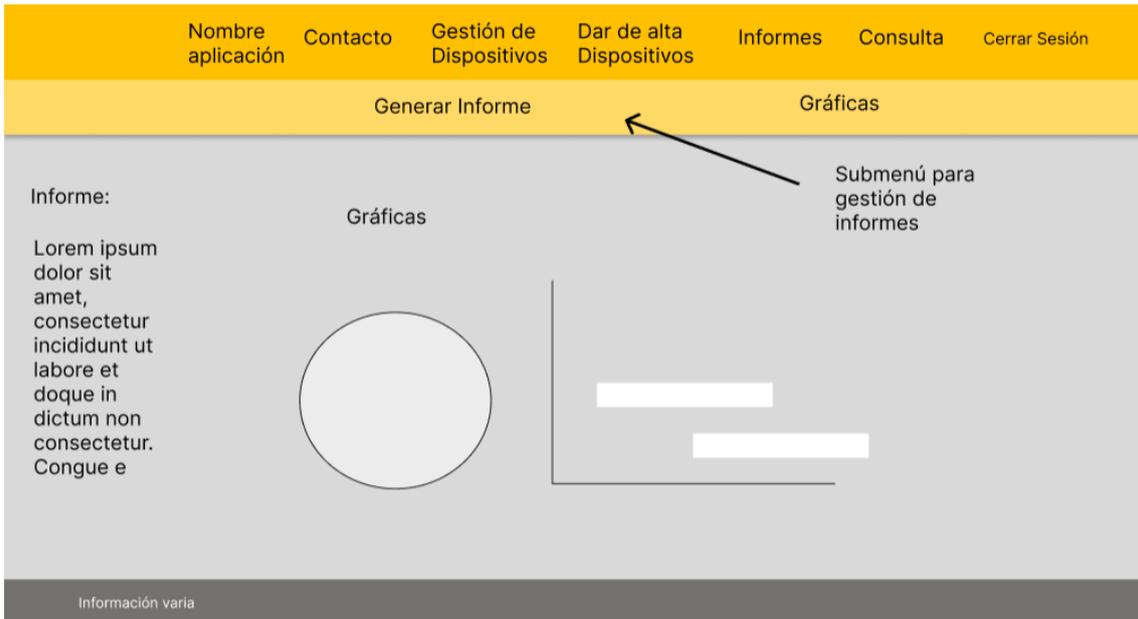


Figura 4.5 Mockup de visualización de informes

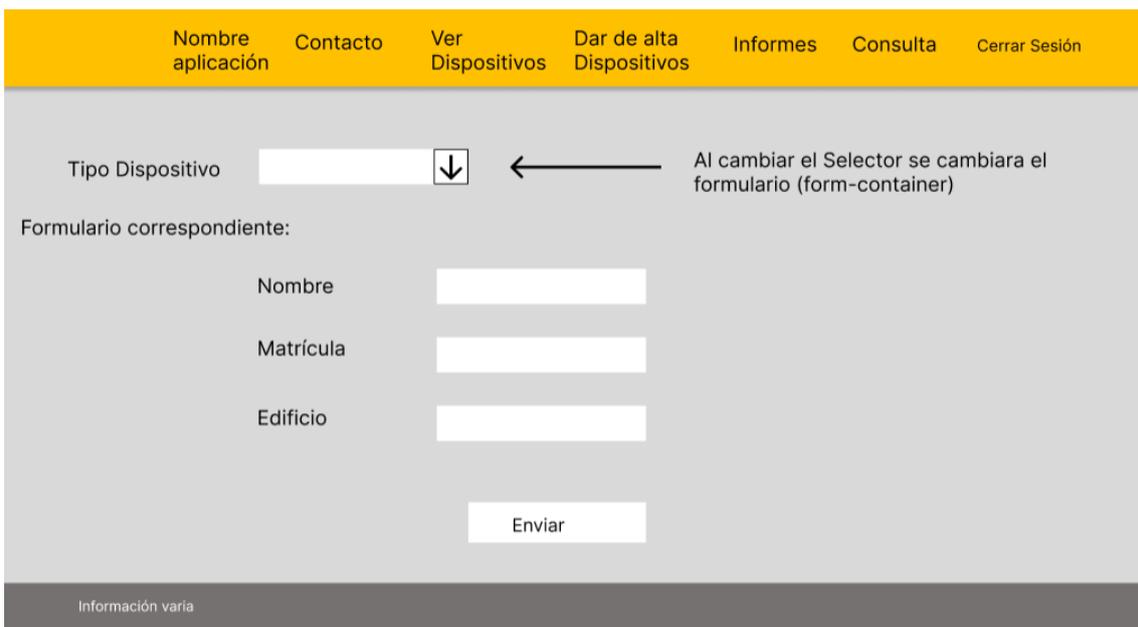


Figura 4.6 Mockup de interfaz de añadir nuevo dispositivo a una empresa

5. Desarrollo de la solución propuesta

Una vez completadas las fases anteriores de especificaciones, análisis y diseño, es momento de empezar el desarrollo del proyecto, aquí se explicará cómo se ha pasado de la propuesta inicial a la solución final, comentando las dificultades y problemas encontrados durante su realización, así como las decisiones que se han tenido que tomar en determinados momentos.

5.1 Módulo de cálculo de la huella de carbono

En primer lugar se desarrolló el módulo de cálculo de huella de carbono, que más tarde sería integrado en la API, por norma general, el cálculo de emisión de un dispositivo se calcula como su dato de consumo (ya sea en litros de combustible, o en kilovatios) multiplicado por su factor de emisión:

$$\text{Huella de carbono} = \text{Dato Consumo} \times \text{Factor Emisión}$$

El dato de consumo será recogido de los datos que el cliente introduzca en la base de datos, y para el factor de emisión se utilizarán las tablas proporcionadas por el Ministerio para la Transición Ecológica del Gobierno de España [12], donde también tienen una calculadora en formato Excel para calcular la huella de carbono de una empresa. Este proyecto utilizará los cálculos usados en esta calculadora Excel, de manera que la rigurosidad del resultado de emisiones sea lo mejor posible. [13]

El resultado de la huella de carbono se medirá en CO2 equivalente (CO2eq), que es la suma de todos los gases de efecto invernadero convertidos a su emisión equivalente en CO2, por ejemplo, una tonelada de Metano (CH4), produce las mismas emisiones que 25 toneladas de CO2, por lo que al pasar todas las unidades a CO2eq, se simplifica el entendimiento de los datos de emisión. [14], [15]

5.2 Creación de la API REST

A continuación se crea la API web con ASP.NET Core en C# gracias al Entity Framework Core, esto nos permitirá poder desplegar con mayor facilidad contenedores en Linux. También nos permitirá el uso de consultas LINQ, una manera de hacer consultas de manera más simple a la base de datos, trabajando con los objetos propios de C# y sin tener que hacer consultas SQL sin procesar.

La API está comunicada con la base de datos en MySQL donde se encontrarán toda la información de las empresas, dispositivos y usuarios. Esta comunicación se realizará mediante consultas LINQ que hemos explicado en el párrafo anterior.

El módulo de cálculo comentado en el apartado anterior estará integrado en la API, es decir, la API realizará llamadas a los métodos de cálculo necesarios para calcular la huella de carbono de cada dispositivo. Se decidió integrar el módulo de cálculo en la API y que se quería que esta pudiese operar de manera totalmente independiente.

5.2.1 Autenticación en la API

La API tiene un middleware basado en autenticación JWT, el JSON Web Token se utiliza para transmitir información de forma segura entre dos partes en formato JSON. Un JWT consta de tres partes separadas por puntos: el encabezado (*header*), la carga útil (*payload*) y la firma (*signature*), ejemplificado en la figura 5.1:

HEADER: ALGORITHM & TOKEN TYPE

```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

PAYLOAD: DATA

```
{  
  "rol": "admin",  
  "id": "1"  
}
```

Figura 5.1 Ejemplo de header y payload de un JWT en json

El encabezado contiene información sobre el tipo de token, en este caso JWT y el algoritmo de firma utilizado, en este caso el HS256. La carga útil o *payload* contiene los datos que deseas transmitir, en este caso el identificador y el rol del usuario. La firma se utiliza para verificar la integridad del token y garantizar su autenticidad.

El proceso de funcionamiento de JWT en la API es el siguiente:

(Como precondition, cabe indicar que el usuario ya está registrado en la plataforma)

1. El cliente hace una llamada a la API a la URL destinada para iniciar sesión, enviando su email y contraseña.
2. La API genera un JWT, que tendrá en el apartado de *payload* el identificador y rol de ese usuario, y el servidor firma el JWT combinando el encabezado y la carga útil con una clave secreta conocida solo por el servidor.
3. La API envía una respuesta HTTP con el JWT al cliente.
4. El cliente recibe el token, y podrá enviarlo en la cabecera de las siguientes peticiones HTTP que realice que requieran autenticación.
5. Cuando la API reciba una petición que lleve un token en la cabecera, comprobará si el token es válido (la firma es correcta) y si no ha expirado aún.

Este proceso nos asegura que el usuario que intenta identificarse es quien dice ser, la validez de los tokens creados es de 7 días, lo que permite al usuario enviar este token en la cabecera de las siguientes peticiones y así no tener que enviar en cada petición sus credenciales. En el caso de la página web la duración de los tokens funcionará de manera distinta, se comentará en el apartado de la creación de la web.

Esto nos permite restringir ciertas llamadas de la API, en nuestro caso no habrá ninguna llamada a la API pública, solamente la de iniciar sesión, y además de poder separar las llamadas dependiendo del rol del usuario, gestionando así los permisos de cada uno.

En el **Anexo IV** se podrán encontrar partes del código interesantes sobre la generación y validación de los tokens JWT.

5.2.2 Llamadas de la API

La creación de llamadas a la API se hace mediante controladores, unas clases que agrupan llamadas sobre un cierto objeto o modelo, un ejemplo de las llamadas que se centran en el usuario son las siguientes:

Usuario	
GET	/api/Organizacion/Usuario
PUT	/api/Organizacion/Usuario
GET	/api/Organizacion/Usuario/{id}
PUT	/api/Organizacion/Usuario/{id}
DELETE	/api/Organizacion/Usuario/{id}
GET	/api/Organizacion/Usuario/all
POST	/api/Organizacion/Usuario/register
POST	/api/Organizacion/Usuario/login

Figura 5.2 Ejemplo de las URL de llamadas para la clase Usuario

Por ejemplo, en este caso la llamada post de inicio de sesión (login) es la única pública, y la de modificar y borrar el usuario por id (la cuarta y la quinta) solamente pueden acceder a ellas aquellos usuarios con el rol de administrador de organización, mientras que el resto son accesibles para todos los usuarios registrados.

Cada llamada de la API tiene el verificador de rol para restringir el acceso si es necesario, y todo tipo de gestión de errores para informar al cliente que está haciendo mal, algunos ejemplos son, modificar o eliminar un dispositivo que no pertenece a la empresa, agregar consumo a un dispositivo que no existe, etc.

Adicionalmente, un factor importante es la seguridad, por lo que se ha implementado un modelo de clases basado en el patrón de diseño DTO (Data Transfer Object), el objetivo de esto en este caso es crear nuevos modelos que únicamente tengan los atributos necesarios. Por ejemplo, la clase Vehículo consta de cinco atributos: Id, matrícula, edificio, categoría del vehículo y Id de la Organización, si se quisiera

modificar el vehículo se creará un objeto que únicamente tenga los atributos de edificio y matrícula, esto evita errores y aumenta la seguridad ya que el usuario simplemente puede modificar esos campos. En la API este sistema se ha implementado con AutoMapper, que ha sido descrito en el apartado 4.3.

En el **Anexo V** se encontrarán todas las llamadas posibles que se pueden realizar a la API.

5.3 Creación de la base de datos

La base de datos se ha creado con MySQL, y todas las tablas y sus columnas se han creado automáticamente con el Entity Framework, el proceso es sencillo, crear los modelos de datos que queramos representar en las tablas.

Para la creación de la base de datos se ha seguido una aproximación llamada "Code First", es un enfoque de desarrollo en el cual se define primero el modelo de datos y las relaciones en el código, y luego Entity Framework se encarga de generar la base de datos correspondiente.

Con Entity Framework Core se pueden ejecutar una serie de comandos para crear unas migraciones que generan el código SQL necesario para crear las tablas y columnas.

Con este comando, a partir de los modelos que hayamos indicado, se generará el código con las modificaciones necesarias a la base de datos, o con el código necesario para su creación si es la primera vez que se ejecuta:

Add-Migration NombreDeLaMigracion

También podemos eliminar una migración antes de aplicarla a la base de datos:

Remove-Migration

Y con el siguiente comando se aplican estos cambios a la base de datos:

Update-Database

Aparte, si se añaden modificaciones a estos modelos, o se crean nuevos, se crea una nueva migración que se encargará de hacer los cambios necesarios a la base de datos conservando los datos que se encuentran en las tablas. Las migraciones son muy útiles ya que ofrecen una forma práctica y automatizada de mantener sincronizada la estructura de la base de datos con el modelo de datos de la aplicación, evitando la necesidad de escribir manualmente el código SQL para crear o modificar tablas.

Adicionalmente, si se desea borrar la base de datos entera lo podemos hacer con el comando:

Drop-Database

5.4 Creación de la página web

Como se ha comentado en el apartado 4.3, la página web se realizará con Blazor, cada página tendrá una URL única, y estará formada por el código HTML y CSS para la interfaz, y código en C# para dar la funcionalidad a los componentes de la interfaz.

Blazor tiene una característica importante que permite recargar los componentes de la página web, llamada *StateHasChanged*, por ejemplo cuando se muestra una lista y se aplica un filtro de búsqueda, en vez de recargar toda la página, este se encarga de recargar únicamente los componentes que han cambiado.

En el HTML Blazor permite usar tags (componentes de la interfaz) adicionales, como por ejemplo el tag *<Virtualize>* que permite renderizar listas de datos enteras a medida que el cliente va navegando hacia abajo en la página, lo que aligera el tiempo de carga drásticamente o el tag *<EditForm>* que permite crear formularios con validación fácilmente.

El funcionamiento del token JWT en la web es diferente, cuando el usuario inicia sesión, el token se guardará en el lado del cliente, esto permitirá que se incluya como cabecera en todas las peticiones sin que el usuario se preocupe por ello. Cuando el usuario cierra sesión el token se destruirá y se convertirá inválido, la sesión se guardará aunque cierre la pestaña o el navegador, hasta un total de la fecha máxima de duración del token, en este caso 7 días.

Por otro lado tenemos todas las clases encargadas de comunicarse con la API, que serializarán los datos introducidos por el cliente para hacer las peticiones.

En esta web se tendrá la posibilidad de añadir, modificar o eliminar dispositivos y sus consumos, visualizar los datos personales, y las listas de todos los dispositivos y consumos.

A continuación se mostrarán ejemplos de algunas interfaces del resultado final de la aplicación:

Web Cálculo Huella de Carbono Dar dispositivo de Alta Añadir Nuevo Consumo Gestión de Dispositivos Informes Empresa Cerrar Sesión

Dar dispositivo de alta

Tipo de dispositivo: Vehiculo

Nuevo Vehículo:

Edificio Sede
Edificio principal

Matricula
2939 ANV

Categoria del Vehiculo
Categoria N1

Añadir Vehículo

Figura 5.3 Interfaz para dar de alta un dispositivo nuevo en el sistema

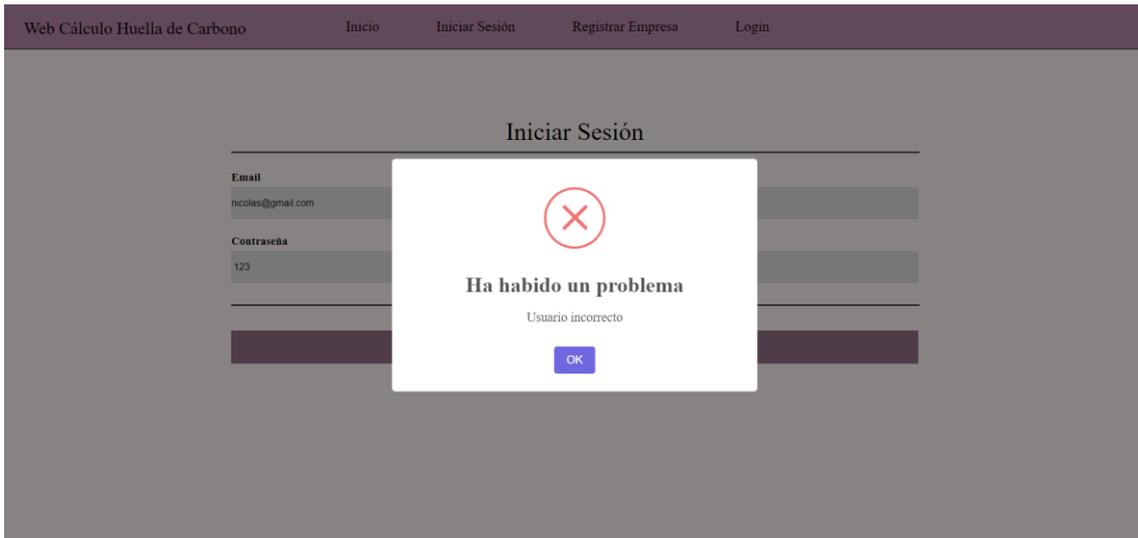


Figura 5.4 Alerta de error al fallar el inicio de sesión



Figura 5.5 Agregar un nuevo consumo a un dispositivo

Web Cálculo Huella de Carbono Dar dispositivo de Alta Añadir Nuevo Consumo Gestión de Dispositivos Informes Empresa Cerrar Sesión

LISTA DE USUARIOS

Filtrar: Email Filtrar

ID USUARIO	NOMBRE Y APELLIDOS	EMAIL	CONTRASEÑA	MODIFICAR	ELIMINAR
4	Mmanuel	Mann@gmail.com	12314	<input type="button" value="M"/>	<input type="button" value="X"/>
5	Alex	alex@gmail.com	12314	<input type="button" value="M"/>	<input type="button" value="X"/>
6	joan	joan@gmail.com	12314	<input type="button" value="M"/>	<input type="button" value="X"/>
7	raul	raul@gmail.com	12314	<input type="button" value="M"/>	<input type="button" value="X"/>
8	Marcos	marcos@gmail.com	12314	<input type="button" value="M"/>	<input type="button" value="X"/>
9	Pere	pere@gmail.com	12314	<input type="button" value="M"/>	<input type="button" value="X"/>
10	Tino	tino@gmail.com	12314	<input type="button" value="M"/>	<input type="button" value="X"/>

Figura 5.6 Visualización de la lista de usuarios de la empresa (vista de administrador)

Web Cálculo Huella de Carbono Dar dispositivo de Alta Añadir Nuevo Consumo Gestión de Dispositivos Informes Cuenta Cerrar Sesión

Gestión de Dispositivos

Tipo de dispositivo: Vehículo

LISTA DE VEHÍCULOS

Filtrar: Categoría de Vehículo Filtrar

ID VEHICULO	MATRÍCULA	CATEGORÍA DEL VEHICULO	EDIFICIO	VER CONSUMOS
12	3233 DWF	M1	edificio principal	<input type="button" value="C"/>
13	1243 DWF	M1	edificio principal	<input type="button" value="C"/>
14	1243 FFF	M1	edificio principal	<input type="button" value="C"/>
15	2243 FFF	M1	edificio principal	<input type="button" value="C"/>
16	2443 FFF	M1	edificio principal	<input type="button" value="C"/>
17	3193 FFF	M1	edificio principal	<input type="button" value="C"/>

Figura 5.7 Visualización de la lista de vehículos de la empresa (vista de usuario corriente)

6. Implantación

En este apartado se explicará cómo se ha llevado la implantación de este proyecto.

Existen dos alternativas entre las que poder elegir:

- La plataforma será instalada en los servidores de los clientes como Centro de Procesamiento de Datos (CPD) interno propiedad del cliente o servidores en la nube, pero administrados por el cliente.

- Que le empresa le ofrezca el servicio de adherirse a la plataforma online bajo su dominio. El cliente paga por el acceso, almacenamiento de sus datos, utilización de la plataforma y mantenimiento de los servicios en un solo pago (puede ser mensual o anual).

Para este proyecto se ha optado por la opción de tener una plataforma online donde los clientes pagan por el servicio, puesto que la aplicación está destinada a ofrecer un servicio en el que almacenar los datos del cliente.

6.1 Selección de proveedores de servicios online

Lo primero que hay que hacer para implantar un servicio online, se debe seleccionar un proveedor, y para abordar la selección de los proveedores se debe seguir una serie de pautas:

1. Evaluación de características y funcionalidades: Comparar las características y funcionalidades ofrecidas por cada proveedor. Se tienen que considerar aspectos como la capacidad de orquestación, el soporte de almacenamiento persistente, la gestión de redes, las herramientas de monitoreo y las integraciones con otros servicios en la nube.

2. Evaluación de la escalabilidad y el rendimiento: Verificar la capacidad de escalabilidad y el rendimiento de los proveedores. Considerar aspectos como el auto escalado, la gestión de clústeres, el rendimiento de la red y la latencia.

3. Evaluación de la seguridad: Analizar las medidas de seguridad ofrecidas por los proveedores. Verificar características como el aislamiento de contenedores, el cifrado de datos, la autenticación y autorización, y las auditorías de seguridad.

4. Evaluación de la disponibilidad y el soporte técnico: Asegurarse de que el proveedor ofrezca una alta disponibilidad y un soporte técnico adecuado. Considerar aspectos como los acuerdos de nivel de servicio (SLA), la capacidad de recuperación ante desastres y la calidad del soporte técnico ofrecido.

5. Análisis de costos: Evaluar los costos asociados con el uso de los servicios de contenerización en la nube. Considerar aspectos como los precios por hora o por uso, los modelos de precios, los costos adicionales por almacenamiento y transferencia de datos, y las opciones de descuento o planes de precios a largo plazo.

6. Evaluación de la comunidad y el ecosistema: Considerar la comunidad y el ecosistema alrededor de cada proveedor. Verificar la disponibilidad de documentación, tutoriales, foros de soporte y la adopción generalizada de la tecnología.

Después de una búsqueda de proveedores, las principales opciones para elegir son:

- Amazon Web Services (AWS): Ofrece Amazon Elastic Container Service (ECS) y Amazon Elastic Kubernetes Service (EKS).
- Microsoft Azure: Proporciona Azure Container Instances (ACI) y Azure Kubernetes Service (AKS).
- Google Cloud Platform (GCP): Ofrece Google Kubernetes Engine (GKE).
- OVH : Donde podemos contratar un servidor y utilizar Docker y Docker Compose para la generación de los servicios contenerizados del proyecto.

Al final se ha seleccionado la opción de contratar un servidor en OVH e instalar la plataforma con sus servicios puesto que es la opción más económica de las opciones anteriores.

6.2 Instalación

Cuando se realiza la contratación del servidor se puede indicar el SO a utilizar, en este caso, se usará el sistema operativo Linux Ubuntu Server 22.04 LTS.

En cuanto a recursos de la máquina seleccionada, dispone de 4 vCPU con 16GB RAM y 160 GB de SSD y la conexión para la configuración se realiza mediante conexión SSH

Pasos que seguir para la instalación del stack tecnológico de la plataforma, junto con los comandos correspondientes a ejecutar en cursiva:

1. Actualización de los paquetes del sistema

```
sudo apt update
```

2. Instalar las dependencias necesarias para agregar repositorios sobre HTTPS:

```
sudo apt install apt-transport-https ca-certificates curl software-properties-common
```

3. Agregar la clave GPG oficial de Docker

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

4. Agregar el repositorio de Docker al sistema:

```
echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

5. Actualizar nuevamente los paquetes del sistema para reflejar los cambios en los repositorios.

```
sudo apt update
```

6. Instalar Docker

```
sudo apt install docker-ce docker-ce-cli containerd.io
```

7. Verificar que Docker se haya instalado correctamente iniciando el servicio de Docker.

```
sudo systemctl start Docker
```

8. Verificar la versión de Docker instalada.

```
docker --version
```

A continuación en la Figura 6.1, para la instalación de los servicios en Docker se muestra el archivo de configuración docker-compose.yml que contiene un servicio API REST de .NET Core, un *Frontend* con Blazor y una base de datos MySQL:

```
1 version: '3.9'
2 services:
3   api:
4     build:
5       context: ./api # Carpeta donde se encuentra el código fuente de la API
6       dockerfile: Dockerfile
7     ports:
8       - 5000:5000 # Puerto del contenedor expuesto hacia el host
9     depends_on:
10      - db
11   frontend:
12     build:
13       context: ./frontend # Carpeta donde se encuentra el código fuente del frontend
14       dockerfile: Dockerfile
15     ports:
16       - 80:80 # Puerto del contenedor expuesto hacia el host
17     depends_on:
18       - api
19   db:
20     image: mysql:latest
21     restart: always
22     environment:
23       MYSQL_ROOT_PASSWORD: example # Contraseña del usuario root
24       MYSQL_DATABASE: mydatabase # Nombre de la base de datos
25       MYSQL_USER: myuser # Nombre de usuario de la base de datos
26       MYSQL_PASSWORD: example # Contraseña del usuario de la base de datos
27     ports:
28       - 3306:3306 # Puerto del contenedor expuesto hacia el host
```

Figura 6.1 Archivo de configuración docker-compose.yml

La estructura de directorios después de instalar todo lo necesario para la implantación será la siguiente:

- docker-compose.yml
- Proyecto de la API/
 - Dockerfile
 - (código fuente de la API)
- Proyecto de *Frontend* /
 - Dockerfile
 - (código fuente del *Frontend*)

Donde se indica (código fuente de la API) y (código fuente del *Frontend*), realmente se situará el proyecto con todos sus archivos y carpetas.

Para ejecutar los contenedores, se abre una terminal en la misma ubicación donde se encuentra el archivo `docker-compose.yml` y se ejecuta el siguiente comando:

```
docker-compose up
```

Esto iniciará los contenedores definidos en el archivo `docker-compose.yml`. La API estará disponible en `http://localhost:5000` y el *Frontend* estará disponible en `http://localhost:80`.

6.3 Pasos extra después de la implantación

Una vez el servicio ya esté implantado y en funcionamiento, se deberán de tener en cuenta ciertos aspectos relacionados con el servicio ofrecido y los clientes que lo contratarán.

Cuando un cliente contrate el servicio, se tendrá que realizar la creación o alta del cliente en la plataforma, así como, añadir los usuarios y roles de acceso a los datos y de la plataforma.

También se tendrá en cuenta la ingesta inicial de dispositivos del cliente, mediante un archivo EXCEL o CSV con un formato que la empresa establecerá, de manera que se listen todos los dispositivos del cliente (vehículos, iluminaria, etc...) y se pueda realizar una ingesta inicial de los dispositivos con los campos requeridos para ser guardados en la base de datos.

En cuanto a la monitorización y soporte de la plataforma se debe tener un control del funcionamiento de la infraestructura de la plataforma, para ello, se puede montar un sistema de monitorización como NAGIOS, ZABBIX, Grafana, etc... Con la monitorización de la plataforma se garantiza un soporte ante incidencias de infraestructura o caída de rendimientos. Se generan alarmas vía Email o servicios de notificaciones Push para que un equipo de soporte 24x7 reciban las incidencias y actúen de manera urgente.

Un aspecto muy importante es el despliegue de nuevas funcionalidades de la plataforma, para ello hay que intentar minimizar el impacto o el tiempo fuera de servicio del sistema mientras se realizan las tareas de actualización o mantenimiento.

Este aspecto se puede solucionar de dos maneras:

- Hacer una ventana de mantenimiento en horario "no crítico", es decir, parar el acceso al entorno, realizar las actualizaciones y arrancar otra vez el sistema.

- Colocando un sistema de balanceo de peticiones HTTP delante del servidor, e subir las actualizaciones a un nuevo servidor y desviar el tráfico cuando esté instalado. Para esta opción, la base de datos debe estar compartida por los servidores. Con esta opción, el tiempo de parada es quasi-nulo.

7. Validación y pruebas

Las pruebas de software tienen como objetivo principal detectar defectos o fallos en el software, asegurando su correcto funcionamiento y cumplimiento de los requisitos establecidos. Desempeñan un papel esencial en el desarrollo de sistemas y aplicaciones, asegurando la calidad y el correcto funcionamiento del software. Mediante la detección y corrección de errores, las pruebas contribuyen a ofrecer productos más confiables, seguros y satisfactorios para los usuarios finales.

7.1 Pruebas funcionales de la API

Por un lado tenemos las pruebas de testeo de la API, que para este caso son las más importantes puesto que es el sistema más importante dentro del proyecto. Para las pruebas de la API se ha utilizado el software Postman que es una herramienta de colaboración y desarrollo de API que permite a los desarrolladores probar, documentar y compartir APIs de manera eficiente. Se presenta como una plataforma integral que facilita la creación de solicitudes HTTP, la automatización de pruebas, la visualización de respuestas y la colaboración en proyectos de desarrollo de API.

En Postman se pueden guardar peticiones para lanzarlas en cualquier momento, se harán testeos a todas las URL de la API probando que resultados pueden provocar error, para luego poder capturarlos en la API y que muestre un mensaje de error acorde a ello. Los errores más comunes a la hora de testear la API es enviar caracteres inválidos, hace peticiones sin token, enviar parámetros que no corresponden, enviar tipos de datos equivocados.

Por ejemplo para probar el inicio de sesión se hace de la siguiente manera:

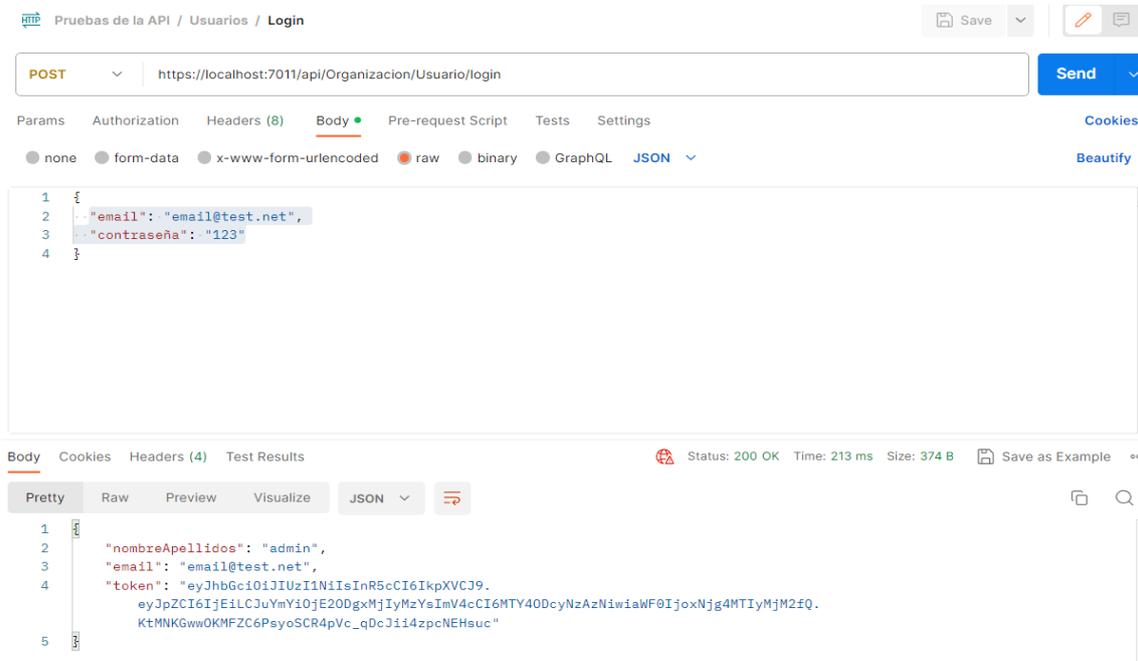


Figura 7.1 Uso de Postman para realizar la petición de inicio de sesión a la API

Se puede observar que introducimos los datos de inicio de sesión y enviamos la petición a la URL de la API y esta nos devuelve el token JWT que se deberá usar para las siguientes peticiones.

Si hacemos una petición sin autorización devuelve los mensajes de errores capturados por la API:

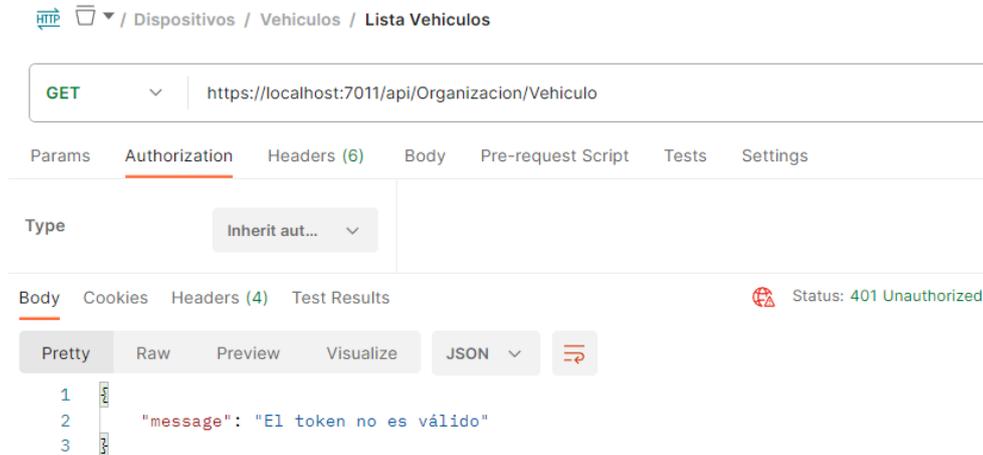


Figura 7.2 Mensaje de error de la API después de una petición fallida

Y por ejemplo, un ejemplo de petición con resultado correcto que devuelve la lista de vehículos de la organización:

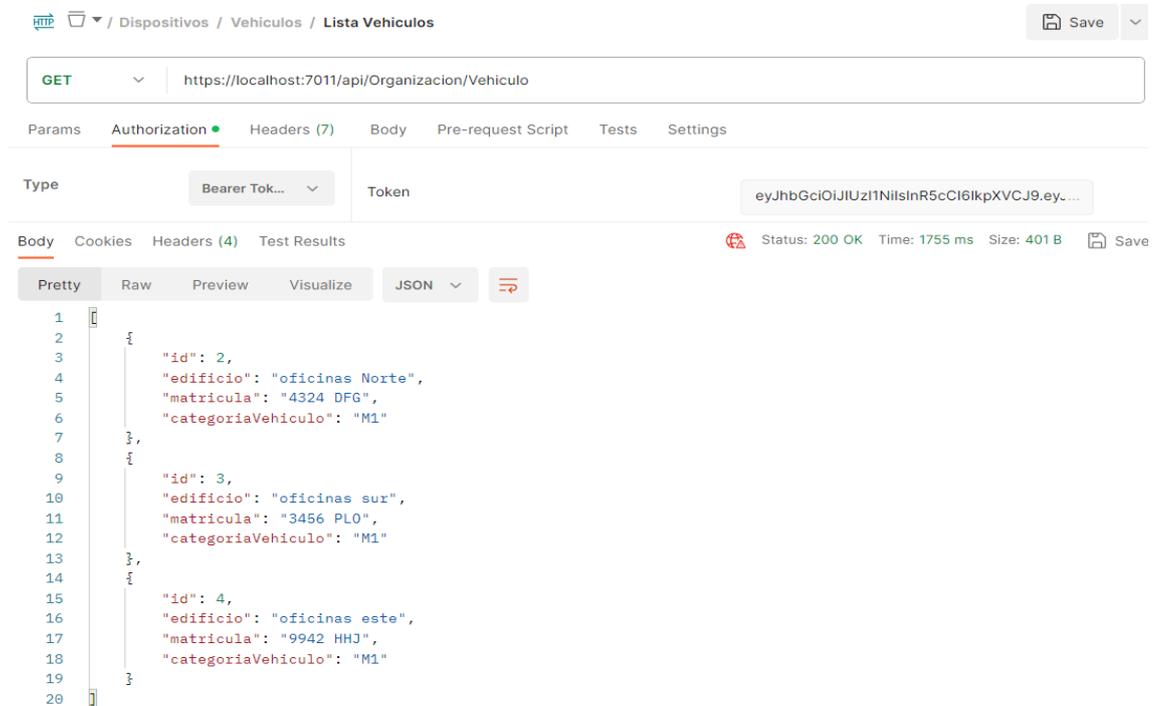


Figura 7.3 Resultado de una petición que devuelve la lista de vehículos de la empresa

A continuación se mostrará en la Figura 7.4 el directorio de carpetas de Postman para realizar las pruebas a la API.

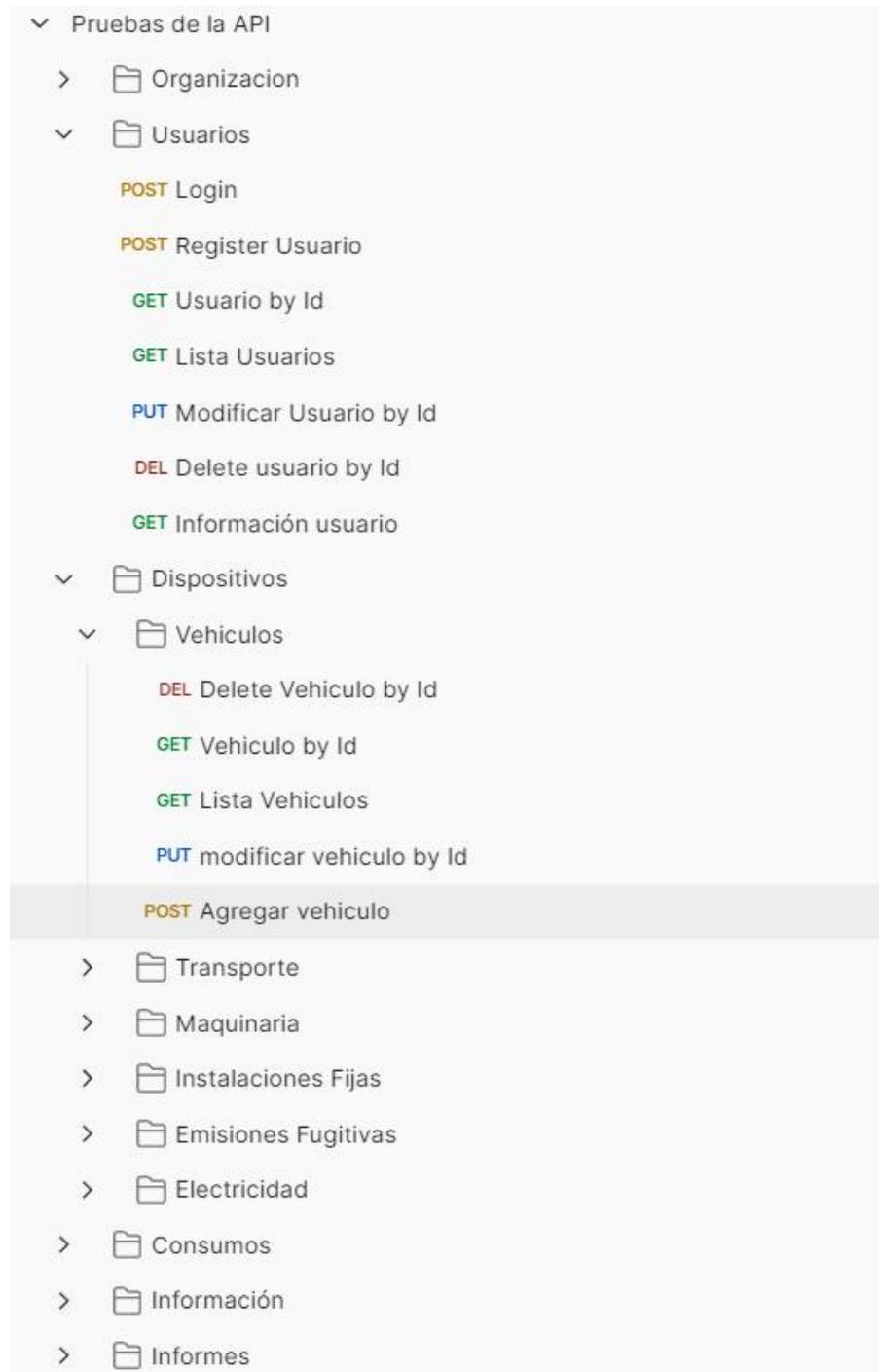


Figura 7.4 Directorio de carpetas con todas las peticiones creadas para probar la API

7.2 Pruebas del sistema web

Para las pruebas del sistema web se tenían que comprobar tres aspectos.

El primero es el correcto funcionamiento de la web a la hora de comunicarse con la API, probando las distintas funciones de la web como agregar dispositivos, verificar que las listas de dispositivos, consumos y usuarios se mostrasen los datos correctamente. También se comprobará el estado de la sesión del usuario, tanto la persistencia de la sesión como el cierre de sesión.

El segundo aspecto se trata de las pruebas de usabilidad que cerciorarse que la interfaz web sea intuitiva y accesible, que no existan errores de navegación entre las diferentes pestañas.

Estos dos primeros aspectos han sido probados por compañeros de la empresa, que tienen mucha experiencia en cuanto a trato con el cliente, y desarrollo de aplicaciones web. Una vez probada la web se notificaron los defectos que se encontraron y se corrigieron.

El tercer aspecto trata las pruebas de rendimiento de la web, en concreto a la hora de navegar entre pestañas y sobre todo a la hora de cargar las diferentes listas por pantalla.

Este último aspecto se ha probado por un lado navegando entre pestañas sin encontrarse ningún tiempo de carga alto y por otro lado generando cientos de datos, en concreto, se crearon cientos de vehículos y se visualizó en la web la velocidad de carga de la lista, que es el tiempo de carga de la página, alrededor de unos 400 ms, lo cual es un tiempo aceptable.

8. Conclusiones

Ahora compararemos los objetivos definidos en el apartado 1.2 con los resultados, el objetivo principal que era la creación de una API web funcional que permitiera calcular la huella de carbono en empresas se ha cumplido, el segundo objetivo, también muy importante era crear una plataforma web que usara la API para poder visualizar los datos de manera más sencilla y también se han cumplido.

Dicho esto, a partir de estos objetivos desarrollados se podrían plantear unos cuantos más, como por ejemplo, perfeccionar la interfaz web, contratando a un diseñador gráfico que hiciera una interfaz mucho más pensada en la experiencia de usuario y el código de la propia API, puesto que hay mejoras que hacer en cuanto a captura de errores. Se podría también mejorar la seguridad en la API por ejemplo para prevenir ataques de *over-posting*, que es cuando alguien realiza muchas peticiones en poco tiempo con la intención de tumbar los servidores, o mejorar el rendimiento de la API cuando se comunica con la base de datos haciendo consultas SQL más precisas y complejas para este fin.

Respecto a las dificultades encontradas durante el desarrollo, en primer lugar, hubo un problema relacionado con la base de datos, en un principio se estaba usando SQL Server de Microsoft, pero era confuso su utilización por lo que el tutor de la empresa propuso hacerlo con MySQL que es lo que utilizan ellos y así se consiguió avanzar más fácilmente.

Otro problema fue la identificación de usuarios en la API, y la autorización, puesto que no se tenía conocimiento sobre estos temas, la solución fue el uso de autenticación mediante JWT.

Un error cometido fue no haber contemplado toda la captura de errores en los métodos de la API, cada llamada de la API puede tener muchas maneras de acabar en error, y no todos estos errores estaban controlados, por lo que una vez la API estuvo finalizada se tuvo que ir método a método añadiendo las capturas de errores necesarias. Para evitar esto se podría haber tenido en cuenta todas posibilidades de lo que podría salir mal, pero con la poca experiencia poseída era poco viable, así que es un error del cual se aprenderá para próximas ocasiones.

Para este proyecto he tenido que aprender la comunicación a base de peticiones HTTP, el uso del framework Blazor para la interfaz y algunas funciones del Entity Framework Core como las consultas LINQ y el uso de métodos para comunicarse con la base de datos.

Además, he aprendido algunas de las buenas prácticas a seguir para la creación de una API y de interfaces web. Con este proyecto he reforzado mis conocimientos en el uso de las herramientas de .NET, he profundizado en el uso del Entity Framework para el *Backend* y he aprendido nuevas funciones para el *Frontend* con Blazor.

8.1 Relación del trabajo desarrollado con los estudios cursados

Este trabajo está ampliamente relacionado con los estudios cursados, en concreto, la rama de Ingeniería de Software.

A lo largo del desarrollo de este proyecto se ha pasado por todas las fases que se estudiaron en asignaturas como PIN (Proyecto de Ingeniería de Software) o PSW (Proceso de Software), como pueden ser la creación de listas de requisitos, la especificación de requisitos, el diseño y arquitectura del sistema, el desarrollo, la implantación y las pruebas.

Además durante el desarrollo hemos trabajado con diferentes tecnologías, tanto de *Frontend*, con la creación de una web con Blazor, como de *Backend*, con la creación de una API en C# y de una base de datos MySQL. También cabe destacar el uso de Docker para la implantación del sistema y la creación de contenedores.

En resumen, se puede ver que hemos profundizado en aquellas materias que se han cursado, ampliando los conocimientos sobre estas, como por ejemplo el desarrollo y gestión de la base de datos o el uso de contenedores con Docker, y aprendiendo algunas nuevas tecnologías, como podría ser el caso de Blazor. Además, hemos aprendido nuevos conceptos como la autenticación en un API con tokens o el uso del almacenamiento local en la web.

Por último destacaré las competencias transversales que he podido poner en práctica para la elaboración de este TFG:

1. **Aplicación y pensamiento crítico**, se hace referencia a fuentes de información de calidad, como otros TFG o informes de resolución de entidades importantes.
2. **Comprensión e integración de conocimientos**, se declara con anterioridad al desarrollo del proyecto todos los objetivos, funcionamiento, características y requerimientos necesarios.
3. **Análisis y resolución de problemas**, el desarrollo es ordenado, y todas las decisiones tomadas se justifican de manera correcta.
4. **Diseño y proyecto**, el alcance del proyecto ha sido cumplido correctamente, y se han propuesto posibles líneas de trabajo para futuros trabajos.
5. **Responsabilidad ética**, medioambiental y profesional, este trabajo está ampliamente ligado con el medio ambiente y respalda su información en entidades importantes, como las Naciones Unidas, GreenHouse Gas o IPCC.
6. **Problemas contemporáneos**, este proyecto abarca unos de los problemas más importantes de la humanidad actualmente, el cambio climático, que está muy relacionado con los Objetivos de Desarrollo y Sostenibilidad (ODS)
7. **Instrumentación específica**, se han justificado la elección de las diferentes herramientas de software, y se muestra su destreza con ellas durante el desarrollo.

9. Trabajos Futuros

En este proyecto, ha habido flecos que no se han podido acabar por cuestiones de tiempo y que serían interesantes:

En cuanto al consumo de electricidad, lo ideal para las empresas es que sus dispositivos envíen cada cierto tiempo, 1 hora por ejemplo su consumo eléctrico. El problema está en que hay dos maneras de hacerlo, la primera es enviando el consumo de ese tramo de 1 hora, el cual está contemplado en la aplicación, y el otro es enviar el total consumido hasta ese momento, es decir, si consume mil kilovatios cada hora, a la tercera hora enviaría tres mil kilovatios totales, lo cual no está contemplado en la aplicación. Esto sería una opción para mejorar este proyecto, puesto que el consumo eléctrico es de los más importantes.

Esta plataforma ayuda a calcular las emisiones de Alcance 1 y 2, pero también sería muy interesante permitir el cálculo de las emisiones de Alcance 3, de manera a perfeccionar el impacto real de la empresa en el medio ambiente.

Una línea de desarrollo que se podría abrir es ampliar el proyecto para calcular la huella de carbono, no solo en empresas, sino también en ayuntamientos o en las explotaciones agrícolas.

Otra línea de desarrollo podría consistir en montar una plataforma que además de calcular la huella de carbono, también proporcione ayuda a la hora de reducir el consumo, creando planes personalizados para cada entidad que lo solicitase.

10. Referencias

- [1] Working Group I Contribution to the IPCC Fifth Assessment Report Climate Change 2013: The Physical Science Basis Summary for Policymakers, página 25 https://www.climatechange2013.org/images/uploads/WGIAR5-SPM_Approved27Sep2013.pdf
- [2] WMO Greenhouse Gas Bulletin. No 1: 14 March 2006 https://library.wmo.int/doc_num.php?explnum_id=10904
- [3] European Climate Pact , Climate Policy, GHG Protocol https://climate-pact.europa.eu/resources/climate-policy/ghg-protocol_en
- [4] GHG Protocol <https://ghgprotocol.org/>
- [5] Acuerdo de París 2015, Naciones Unidas, Artículo 2 https://unfccc.int/sites/default/files/spanish_paris_agreement.pdf
- [6] Perelló Guarro, PI. (2014). Cálculo de la huella de carbono en el engorde de la lubina. <http://hdl.handle.net/10251/53318>
- [7] Sendra Domenech, C. (2021). Huella de carbono en una industria de bebidas. Universitat Politècnica de València. <http://hdl.handle.net/10251/173649>
- [8] Ruiz Castillo, A. (2020). CÁLCULO DE LA HUELLA DE CARBONO DE UNA INDUSTRIA CERVECERA. Universitat Politècnica de València. <http://hdl.handle.net/10251/157904>
- [9] Bravo Villar, MC. (2022). Análisis comparativo de modelos de reducción de emisiones de CO2 en el transporte marítimo. Universitat Politècnica de València. <http://hdl.handle.net/10251/187075>
- [10] Villanueva Párraga, D. (2021). Análisis energético y de emisiones de dióxido de carbono equivalentes del municipio de Sagunto entre los años 2010 y 2019. Definición y evaluación de las propuestas a implementar para el cumplimiento del Pacto de los Alcaldes para el Clima y la Energía. Universitat Politècnica de València. <http://hdl.handle.net/10251/168144>
- [11] Documentación de Microsoft sobre Blazor de ASP.NET Core <https://learn.microsoft.com/es-es/aspnet/core/blazor/?view=aspnetcore-7.0#blazor-server>
- [12] Factores de emisión emitidos por el Ministerio para la Transacción Ecológica. Año 2022 https://www.miteco.gob.es/es/cambio-climatico/temas/mitigacion-politicas-y-medidas/factoresemission_tcm30-479095.pdf
- [13] Calculadoras del Ministerio para la Transacción Ecológica <https://www.miteco.gob.es/es/cambio-climatico/temas/mitigacion-politicas-y-medidas/calculadoras.aspx>
- [14] Tabla de potencial de calentamiento global de GEI más importantes. Página 53 https://ghgprotocol.org/sites/default/files/standards_supporting/GHGP_GPC%20%28Spanish%29.pdf
- [15] Report of the Conference of the Parties on its nineteenth session, held in Warsaw from 11 to 23 November 2013. Página 24 <https://unfccc.int/resource/docs/2013/cop19/eng/10a03.pdf>

Anexo I: Objetivos de desarrollo sostenible

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

Objetivos de Desarrollo Sostenibles	Alto	Medio	Bajo	No Procede
1. Fin de la pobreza.				X
2. Hambre cero.				X
3. Salud y bienestar.				X
4. Educación de calidad.				X
5. Igualdad de género.				X
6. Agua limpia y saneamiento.				X
7. Energía asequible y no contaminante.		X		
8. Trabajo decente y crecimiento económico.				X
9. Industria, innovación e infraestructuras.				X
10. Reducción de las desigualdades.				X
11. Ciudades y comunidades sostenibles.				X
12. Producción y consumo responsables.	X			
13. Acción por el clima.	X			
14. Vida submarina.				X
15. Vida de ecosistemas terrestres.		X		
16. Paz, justicia e instituciones sólidas.				X
17. Alianzas para lograr objetivos.				X

De los anteriores objetivos de desarrollo sostenible de la tabla anterior, este proyecto está relacionado con:

- **Acción por el clima**, en mi opinión este proyecto es un aporte más para la lucha contra el cambio climático, ya es una realidad que el planeta va en mal camino, por eso, proyectos como este están enfocados a facilitar que las organizaciones se den cuenta realmente del impacto que tienen sobre el clima, y que puedan tomar medidas al respecto para dejar el planeta en el mejor estado posible para las futuras generaciones.
- **Producción y consumo responsables**, este proyecto servirá a las organizaciones para calcular su huella de carbono y, por lo tanto, visualizar su producción y consumo de energía, gracias a esto podrán ver que aparatos o maquinarias son las que más consumen de la empresa y podrían optar por

reemplazarlas por material menos contaminante o incluso buscar otros métodos que sean menos nocivos para el medio ambiente. Esto fomenta el uso de energías renovables para que el uso de energía de una organización no dependa solamente de las comercializadoras de energía no renovables.

- **Vida de ecosistemas terrestres**, lo que se busca en este proyecto es la reducción de la huella de carbono por parte de las empresas, el cambio climático provoca cambios en el ecosistema, ya que aumentan las temperaturas, esto provoca el deshielo en las zonas heladas y la sequía en los lugares más cálidos. A su vez, la destrucción del ecosistema contribuye al empeoramiento del cambio climático como por ejemplo la deforestación, por lo que tenemos que cuidar los ecosistemas directamente, por ejemplo, no deforestando tanto como hoy en día, y también indirectamente intentando luchar contra el cambio climático reduciendo nuestra huella de carbono.
- **Energía asequible y no contaminante**, como ya hemos comentado en el apartado de producción y consumo responsable, el objetivo es incitar a las empresas a que contraten su energía de fuentes renovables, con el fin de intentar contaminar lo menos posible y de reducir su impacto con el medio ambiente.

Anexo II: Descripción de Casos de Uso

Id y Nombre	CU-1 Iniciar Sesión
Descripción	Permite iniciar sesión en el sistema
Precondición	No
Secuencia Principal	1. Introducir nombre y contraseña del usuario
Alternativas / Errores	1.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	2. El sistema web registrará el token del usuario para utilizarlo en las cabecera de las siguientes peticiones a la API.

Id y Nombre	CU-3 Dar dispositivo de Alta
Descripción	Permite a cualquier usuario crear un nuevo dispositivo en su empresa
Precondición	Haber iniciado sesión en el sistema
Secuencia Principal	1. Enviar datos del dispositivo a crear
Alternativas / Errores	1.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	2. El sistema creará el dispositivo y lo asignará a la empresa

Id y Nombre	CU-4 Añadir nuevo consumo de un dispositivo
Descripción	Se añadirá un consumo a un dispositivo de la empresa
Precondición	Haber iniciado sesión en el sistema
Secuencia Principal	1. Introducir datos del consumo de un dispositivo a crear
Alternativas / Errores	1.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	2. El sistema creará el consumo del dispositivo y lo asignará a la empresa

Id y Nombre	CU-5 Ver información de la cuenta
Descripción	El usuario verá la información relacionada a su empresa y a su usuario
Precondición	Haber iniciado sesión en el sistema
Secuencia Principal	1. Seleccionar la pestaña de ver la información de la cuenta
Alternativas / Errores	1.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	2. El sistema mostrará la información pedida

Id y Nombre	CU-6 Ver Lista de Dispositivos
Descripción	El usuario visualizará la lista de los dispositivos por tipo
Precondición	Haber iniciado sesión en el sistema
Secuencia Principal	1. Seleccionar la pestaña de visualizar la lista de dispositivos, indicando el tipo de dispositivo del cual se quiere mostrar
Alternativas / Errores	1.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.

Post condición	2. El sistema mostrará la lista de dispositivos pedida
----------------	--

Id y Nombre	CU-7 Ver lista de consumos
Descripción	El usuario visualizará la lista de los consumos de un dispositivo por tipo
Precondición	Haber iniciado sesión en el sistema
Secuencia Principal	1. Seleccionar la pestaña de visualizar la lista de consumos de un dispositivo, indicando el tipo de dispositivo del cual quiere ver la lista
Alternativas / Errores	1.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	2. El sistema mostrará la lista de consumos de dispositivos pedida

Id y Nombre	CU-8 Eliminar Dispositivo
Descripción	Elige un dispositivo de la empresa para borrarlo
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización
Secuencia Principal	1. Seleccionar la lista de dispositivos 2. Presionar el botón de eliminar correspondiente al dispositivo deseado
Alternativas / Errores	2.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	3. El sistema elimina el dispositivo elegido

Id y Nombre	CU-9 Modificar dispositivo
Descripción	Elige un dispositivo de la empresa para modificarlo
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización
Secuencia Principal	1. Seleccionar la lista de dispositivos 2. Presionar el botón de modificar correspondiente al dispositivo deseado 3. Modificar los parámetros deseados
Alternativas / Errores	3.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	4. El sistema modifica el dispositivo elegido

Id y Nombre	CU-10 Modificar Consumo
Descripción	Elige el consumo de un dispositivo de la empresa para modificarlo
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización
Secuencia Principal	1. Seleccionar la lista de consumos de dispositivos 2. Presionar el botón de modificar correspondiente al consumo deseado 3. Modificar los parámetros deseados
Alternativas / Errores	3.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	4. El sistema modifica el consumo elegido

Id y Nombre	CU-11 Eliminar Consumo
Descripción	Elige el consumo de un dispositivo de la empresa para borrarlo
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización
Secuencia Principal	1. Seleccionar la lista de consumos de dispositivos 2. Presionar el botón de eliminar correspondiente al consumo deseado
Alternativas / Errores	2.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	3. El sistema elimina el consumo elegido

Id y Nombre	CU-12 Ver lista de usuarios
Descripción	Permite visualizar la lista de usuarios agregados en esta empresa
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización
Secuencia Principal	1. Seleccionar la pestaña de visualización de la lista de usuarios de la empresa
Alternativas / Errores	1.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	2. El sistema mostrará la información pedida

Id y Nombre	CU-13 Modificar Usuario
Descripción	Elige un usuario de la empresa para modificarlo
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización
Secuencia Principal	1. Seleccionar la lista de usuarios 2. Presionar el botón de modificar correspondiente al usuario deseado 3. Modificar los parámetros deseados
Alternativas / Errores	3.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	4. El sistema modifica el usuario elegido

Id y Nombre	CU-14 Eliminar Usuario
Descripción	Elige un usuario de la empresa para eliminarlo
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización
Secuencia Principal	1. Seleccionar la lista de usuarios 2. Presionar el botón de eliminar correspondiente al usuario deseado
Alternativas / Errores	2.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	3. El sistema elimina el usuario elegido

Id y Nombre	CU-15 Crear Usuario
Descripción	Crea un nuevo usuario en la empresa
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización

Secuencia Principal	1. Introducir datos del nuevo usuario a crear
Alternativas / Errores	1.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	2. El sistema creará un nuevo usuario

Id y Nombre	CU-16 Modificar Organización
Descripción	Se modificarán los parámetros de la organización
Precondición	Haber iniciado sesión en el sistema con el rol de Administrador de la organización
Secuencia Principal	1. Seleccionar pestaña de modificar organización 2. Modificar parámetros deseados
Alternativas / Errores	2.1. Si sucede un error el sistema web mostrará una alerta indicando el problema.
Post condición	3. El sistema modificará la organización

Anexo III: Clases y atributos

Clases de los modelos de usuarios, organización y dispositivos

Clase	Atributo	Descripción
Organización	Id	Identificador de la empresa
	NombreOrg	En qué lugar de la empresa se encuentra o pertenece
	Email	Email para identificarse
	Dirección	Dirección de la organización
	Contraseña	Contraseña de la Organización
Usuario	Id	Identificador del usuario
	NombreApellidos	En qué lugar de la empresa se encuentra o pertenece
	Email	Email para identificarse
	Contraseña	Contraseña del usuario
	Role	Rol del usuario
	OrganizacionId	Referencia de la organización a la que pertenece
Electricidad	Id	Identificador del dispositivo eléctrico
	Edificio	En qué lugar de la empresa se encuentra o pertenece
	Dispositivo	Nombre del dispositivo o tipo
	Descripción	Descripción del dispositivo
	OrganizacionId	Referencia de la organización a la que pertenece
EmisionesFugitivas	Id	Identificador del equipo de climatización o refrigeración
	Edificio	En qué lugar de la empresa se encuentra o pertenece
	NombreEquipo	Nombre del equipo
	OrganizacionId	Referencia de la organización a la que pertenece
InstalacionesFijas	Id	Identificador de la instalación fija
	Edificio	En qué lugar de la empresa se encuentra o pertenece
	Nombre	Nombre de la instalación fija
	OrganizacionId	Referencia de la organización a la que pertenece
Maquinaria	Id	Identificador de la maquinaria
	Edificio	En qué lugar de la empresa se encuentra o pertenece
	Nombre	Nombre de la maquinaria
	TipoMaquinaria	Tipo de la maquinaria (agrícola, industrial, ...)

	OrganizacionId	Referencia de la organización a la que pertenece
Vehículo	Id	Identificador del vehículo
	Edificio	En qué lugar de la empresa se encuentra o pertenece
	Matrícula	Matrícula del Vehículo
	CategoriaVehiculo	Categoría del vehículo (N1,N2,N3,M1,M2,M3,L)
	OrganizacionId	Referencia de la organización a la que pertenece
Transporte	Id	Identificador del transporte
	Edificio	En qué lugar de la empresa se encuentra o pertenece
	TipoTransporte	Tipo de transporte (aéreo, ferrocarril, marítimo)
	CombustibleTransporte	Combustible del transporte
	OrganizacionId	Referencia de la organización a la que pertenece

Clases de los consumos de los dispositivos

ElectricidadConsumo	Id	Identificador del consumo de un dispositivo eléctrico
	KwH	Consumo del dispositivo en kilovatios
	ComercializadorId	Id de la Comercializadora de electricidad
	Comercializadora	Nombre de la Comercializadora de electricidad
	FechaInicio	Fecha de Inicio del Consumo
	FechaFin	Fecha de Finalización del Consumo
	Consumo	Emisión calculada en CO2 equivalente (huella de carbono)
	ElectricidadId	Id del dispositivo eléctrico al que pertenece este consumo
EmisionesFugitivasConsumo	Id	Identificador del consumo de un equipo de climatización
	Gas	Nombre del gas usado para la recarga
	Recarga	Cantidad de recarga en kilos de gas
	FechaInicio	Fecha de Inicio del Consumo
	FechaFin	Fecha de Finalización del Consumo
	Consumo	Emisión calculada en CO2 equivalente (huella de carbono)
	EmisionesFugitivasId	Id del equipo de climatización al que pertenece este consumo
InstalacionesFijasConsumo	Id	Identificador del consumo de la instalación fija
	CantidadCombustible	Cantidad de combustible consumido

	TipoCombustible	Tipo de combustible de la instalación fija
	FechaInicio	Fecha de Inicio del Consumo
	FechaFin	Fecha de Finalización del Consumo
	Consumo	Emisión calculada en CO2 equivalente (huella de carbono)
	InstalacionFijasId	Id de la instalación fija a la que pertenece este consumo
MaquinariaConsumo	Id	Identificador del consumo de la maquinaria
	CantidadCombustible	Cantidad de combustible consumido
	TipoCombustible	Tipo de combustible de la maquinaria
	FechaInicio	Fecha de Inicio del Consumo
	FechaFin	Fecha de Finalización del Consumo
	Consumo	Emisión calculada en CO2 equivalente (huella de carbono)
	MaquinariaId	Id de la maquinaria a la que pertenece este consumo
TransporteConsumo	Id	Identificador del consumo del transporte
	CantidadCombustible	Cantidad de combustible consumido
	FechaInicio	Fecha de Inicio del Consumo
	FechaFin	Fecha de Finalización del Consumo
	Consumo	Emisión calculada en CO2 equivalente (huella de carbono)
	TransporteId	Id del transporte al que pertenece este consumo
VehiculoConsumo	Id	Identificador del consumo del vehículo
	CantidadCombustible	Cantidad de combustible consumido
	TipoCombustible	Tipo de combustible del vehículo (B5,B7,E5,100, ...)
	FechaInicio	Fecha de Inicio del Consumo
	FechaFin	Fecha de Finalización del Consumo
	Consumo	Emisión calculada en CO2 equivalente (huella de carbono)
	VehiculoId	Id del vehículo a la que pertenece este consumo
OtrosConsumos	Id	Identificador del consumo proveniente de otras fuentes
	Edificio	En qué lugar de la empresa se encuentra o pertenece
	Nombre	Nombre del consumo
	CategoriaActividad	A que sector pertenece este consumo
	CantidadConsumo	Cantidad consumida

	FactorEmision	Factor de emisión de en relación al combustible o gas consumido
	FechaInicio	Fecha de Inicio del Consumo
	FechaFin	Fecha de Finalización del Consumo
	Consumo	Emisión calculada en CO2 equivalente (huella de carbono)
	OrganizacionId	Referencia de la organización a la que pertenece

Tablas maestras con los factores de emisión

FactorEmisionElectricidad	Id	Id de la comercializadora de electricidad
	Comercializadora	Nombre de la comercializadora de electricidad
	Factor	factor de emisión de cada comercializadora
FactorEmisionElectricidad	Id	Id de la comercializadora de electricidad
	Comercializadora	Nombre de la comercializadora de electricidad
	Factor	factor de emisión de cada comercializadora
FactorEmisionMaquinaria	Tipo	Tipo de combustible
	Industrial	Factor de emisión para el tipo de maquinaria industrial
	Agricola	Factor de emisión para el tipo de maquinaria agrícola
	Forestal	Factor de emisión para el tipo de maquinaria forestal
FactorEmisionTransporte	Tipo	Tipo de combustible
	Ferrocarril	Factor de emisión para el tipo de transporte ferrocarril
	Maritimo	Factor de emisión para el tipo de transporte marítimo
	Aereo	Factor de emisión para el tipo de transporte aéreo
FactorEmisionTransporte	Tipo	Tipo de combustible
	M1	Factor de emisión para el tipo de vehículo M1
	N1	Factor de emisión para el tipo de vehículo N1
	N2N3M2M3	Factor de emisión para los tipo de vehículo N2, N3, M2, M3
	L	Factor de emisión para el tipo de vehículo L
FactorEmisionesFugitivas	Nombre	Nombre del gas
	FormulaQuimica	Fórmula química del gas
	PCA	Factor de emisión del gas
	Tipo	Tipo de combustible

FactorEmisionesFugitivas	Factor	Factor de emisión para cada tipo de combustible
--------------------------	--------	---

Anexo IV: Código generación y Validación token JWT

```
public int? ValidateToken(string token)
{
    if (token == null)
        return null;

    var tokenHandler = new JwtSecurityTokenHandler();
    var key = Encoding.ASCII.GetBytes(_appSettings.Secret);
    try
    {
        tokenHandler.ValidateToken(token, new TokenValidationParameters
        {
            ValidateIssuerSigningKey = true,
            IssuerSigningKey = new SymmetricSecurityKey(key),
            ValidateIssuer = false,
            ValidateAudience = false,
            // set clocks skew to zero so tokens expire exactly at token expiration time (instead of 5 minutes later)
            ClockSkew = TimeSpan.Zero
        }, out SecurityToken validatedToken);

        var jwtToken = (JwtSecurityToken)validatedToken;
        var userId = int.Parse(jwtToken.Claims.First(x => x.Type == "id").Value);

        // return user id from JWT token if validation successful
        return userId;
    }
    catch
    {
        // return null if validation fails
        return null;
    }
}
```

Figura 11.1 Código de validación de un token JWT

```
public string GenerateToken(Usuario user)
{
    // generate token that is valid for 7 days
    var tokenHandler = new JwtSecurityTokenHandler();
    var key = Encoding.ASCII.GetBytes(_appSettings.Secret);
    var tokenDescriptor = new SecurityTokenDescriptor
    {
        Subject = new ClaimsIdentity(new[] { new Claim("id", user.Id.ToString()) }),
        Expires = DateTime.UtcNow.AddDays(7),
        SigningCredentials = new SigningCredentials(new SymmetricSecurityKey(key), SecurityAlgorithms.HmacSha256Signature),
    };
    var token = tokenHandler.CreateToken(tokenDescriptor);
    return tokenHandler.WriteToken(token);
}
```

Figura 11.2 Código de generación de un token JWT

Anexo V: Llamadas a la API

A continuación se muestran todas las llamadas a la API agrupadas por las clases con las que tienen relación.

Electricidad	
GET	/api/Organizacion/Electricidad
POST	/api/Organizacion/Electricidad
GET	/api/Organizacion/Electricidad/{id}
PUT	/api/Organizacion/Electricidad/{id}
DELETE	/api/Organizacion/Electricidad/{id}
ElectricidadConsumo	
GET	/api/Organizacion/Electricidad/Consumo
POST	/api/Organizacion/Electricidad/Consumo
GET	/api/Organizacion/Electricidad/Consumo/id/{id}
GET	/api/Organizacion/Electricidad/Consumo/{electricidadid}
PUT	/api/Organizacion/Electricidad/Consumo/{id}
DELETE	/api/Organizacion/Electricidad/Consumo/{id}

Figura 11.3 URL de las llamadas sobre las clases Electricidad y su consumo

InstalacionesFijas	
GET	/api/Organizacion/InstalacionesFijas
POST	/api/Organizacion/InstalacionesFijas
GET	/api/Organizacion/InstalacionesFijas/{id}
PUT	/api/Organizacion/InstalacionesFijas/{id}
DELETE	/api/Organizacion/InstalacionesFijas/{id}
InstalacionesFijasConsumo	
GET	/api/Organizacion/InstalacionesFijas/Consumo
POST	/api/Organizacion/InstalacionesFijas/Consumo
GET	/api/Organizacion/InstalacionesFijas/Consumo/id/{id}
GET	/api/Organizacion/InstalacionesFijas/Consumo/{instalacionesFijaid}
PUT	/api/Organizacion/InstalacionesFijas/Consumo/{id}
DELETE	/api/Organizacion/InstalacionesFijas/Consumo/{id}

Figura 11.4 URL de las llamadas sobre las clases Instalación Fija y su consumo

Maquinaria

GET /api/Organizacion/Maquinaria

POST /api/Organizacion/Maquinaria

GET /api/Organizacion/Maquinaria/{id}

PUT /api/Organizacion/Maquinaria/{id}

DELETE /api/Organizacion/Maquinaria/{id}

MaquinariaConsumo

GET /api/Organizacion/Maquinaria/Consumo

POST /api/Organizacion/Maquinaria/Consumo

GET /api/Organizacion/Maquinaria/Consumo/id/{id}

GET /api/Organizacion/Maquinaria/Consumo/{maquinariaid}

PUT /api/Organizacion/Maquinaria/Consumo/{id}

DELETE /api/Organizacion/Maquinaria/Consumo/{id}

Figura 11.5 URL de las llamadas sobre las clases Maquinaria y su consumo

Organizacion

GET	/api/Organizacion
PUT	/api/Organizacion
DELETE	/api/Organizacion
POST	/api/Organizacion/register

Figura 11.6 URL de las llamadas sobre la Organización

OtrosConsumos

GET	/api/Organizacion/OtrosConsumos
POST	/api/Organizacion/OtrosConsumos
GET	/api/Organizacion/OtrosConsumos/id/{id}
GET	/api/Organizacion/OtrosConsumos/{otrosconsumosid}
PUT	/api/Organizacion/OtrosConsumos/{id}
DELETE	/api/Organizacion/OtrosConsumos/{id}

Figura 11.7 URL de las llamadas sobre los consumos de otras fuentes

Transporte

GET /api/Organizacion/Transporte

POST /api/Organizacion/Transporte

GET /api/Organizacion/Transporte/{id}

PUT /api/Organizacion/Transporte/{id}

DELETE /api/Organizacion/Transporte/{id}

TransporteConsumo

GET /api/Organizacion/Transporte/Consumo

POST /api/Organizacion/Transporte/Consumo

GET /api/Organizacion/Transporte/Consumo/id/{id}

GET /api/Organizacion/Transporte/Consumo/{transporteid}

PUT /api/Organizacion/Transporte/Consumo/{id}

DELETE /api/Organizacion/Transporte/Consumo/{id}

Figura 11.8 URL de las llamadas sobre las clases Transporte y su consumo

Vehiculo

GET /api/Organizacion/Vehiculo

POST /api/Organizacion/Vehiculo

GET /api/Organizacion/Vehiculo/{id}

PUT /api/Organizacion/Vehiculo/{id}

DELETE /api/Organizacion/Vehiculo/{id}

VehiculoConsumo

GET /api/Organizacion/Vehiculo/Consumo

POST /api/Organizacion/Vehiculo/Consumo

GET /api/Organizacion/Vehiculo/Consumo/id/{id}

GET /api/Organizacion/Vehiculo/Consumo/{vehiculoid}

PUT /api/Organizacion/Vehiculo/Consumo/{id}

DELETE /api/Organizacion/Vehiculo/Consumo/{id}

Figura 11.9 URL de las llamadas sobre las clases Vehículo y su consumo

Usuario

GET /api/Organizacion/Usuario

PUT /api/Organizacion/Usuario

GET /api/Organizacion/Usuario/{id}

PUT /api/Organizacion/Usuario/{id}

DELETE /api/Organizacion/Usuario/{id}

GET /api/Organizacion/Usuario/all

POST /api/Organizacion/Usuario/register

POST /api/Organizacion/Usuario/login

Figura 11.10 URL de las llamadas sobre los usuarios

Informacion

GET /api/Informacion/combustiblevehiculo

GET /api/Informacion/combustibletransporte

GET /api/Informacion/combustiblemaquinaria

GET /api/Informacion/combustibleinstalacionesfijas

Figura 11.11 URL de las llamadas de información de valores permitidos

InformesElectricidad

GET /api/Informes/Electricidad

GET /api/Informes/Electricidad/{id}

GET /api/Informes/Electricidad/{id}/mes

InformesEmisionesFijas

GET /api/Informes/EmisionesFijas

GET /api/Informes/EmisionesFijas/{id}

GET /api/Informes/EmisionesFijas/{id}/mes

InformesInstalacionesFijas

GET /api/Informes/InstalacionesFijas

GET /api/Informes/InstalacionesFijas/{id}

GET /api/Informes/InstalacionesFijas/{id}/mes

Figura 11.12 URL de las llamadas de informes, parte 1

InformesMaquinaria

GET /api/Informes/Maquinaria

GET /api/Informes/Maquinaria/{id}

GET /api/Informes/Maquinaria/{id}/mes

InformesOtrosConsumos

GET /api/Informes/OtrosConsumos

GET /api/Informes/OtrosConsumos/{id}

GET /api/Informes/OtrosConsumos/{id}/mes

Figura 11.13 URL de las llamadas de informes, parte 2

Informes Transporte

GET /api/Informes/Transporte

GET /api/Informes/Transporte/{id}

GET /api/Informes/Transporte/{id}/mes

Informes Vehiculo

GET /api/Informes/Vehiculo

GET /api/Informes/Vehiculo/{id}

GET /api/Informes/Vehiculo/{id}/mes

Figura 11.14 URL de las llamadas de informes, parte 3

EmisionesFugitivas

GET /api/Organizacion/EmisionesFugitivas

POST /api/Organizacion/EmisionesFugitivas

GET /api/Organizacion/EmisionesFugitivas/{id}

PUT /api/Organizacion/EmisionesFugitivas/{id}

DELETE /api/Organizacion/EmisionesFugitivas/{id}

EmisionesFugitivasConsumo

GET /api/Organizacion/EmisionesFugitivas/Consumo

POST /api/Organizacion/EmisionesFugitivas/Consumo

GET /api/Organizacion/EmisionesFugitivas/Consumo/id/{id}

GET /api/Organizacion/EmisionesFugitivas/Consumo/{emisionesid}

PUT /api/Organizacion/EmisionesFugitivas/Consumo/{id}

DELETE /api/Organizacion/EmisionesFugitivas/Consumo/{id}

Figura 11.15 URL de las llamadas sobre las clases EmisionesFugitivas y su consumo

Glosario

En este apartado se explicarán algunos términos usados en repetidas ocasiones en la memoria:

- **Frontend:** se refiere a todo lo que los usuarios ven y con lo que interactúan directamente. Se encarga de la presentación visual y la interacción del usuario. El objetivo principal del frontend es proporcionar una interfaz de usuario atractiva y funcional. En este proyecto el frontend hace referencia a la página web.
- **Backend:** se refiere a la parte de la aplicación que no es visible para los usuarios. Se encarga de la lógica del negocio, el almacenamiento de datos y la comunicación con otros sistemas. En este proyecto el backend corresponde al sistema de la API y base de datos.
- **API (Application Programming Interface):** una API es un conjunto de reglas y protocolos que permiten a diferentes aplicaciones comunicarse y compartir datos entre sí de manera estandarizada, en este proyecto el rol de la API es acceder y manipular datos en una base de datos.
- **Docker:** permite a los desarrolladores y administradores de sistemas crear, implementar y ejecutar aplicaciones de forma aislada en contenedores. Un contenedor es una unidad de software liviana y autónoma que contiene todo lo necesario para ejecutar una aplicación, incluyendo el código, las bibliotecas, las dependencias y las configuraciones. Docker proporciona una abstracción a nivel de sistema operativo, lo que significa que los contenedores pueden ejecutarse de manera consistente en diferentes entornos, como computadoras locales, servidores en la nube o clústeres de servidores. Esto facilita la creación y distribución de aplicaciones.
- **JWT (Json Web Token):** es un estándar abierto (RFC 7519) para la creación de tokens de seguridad en formato JSON. Se utiliza para autenticar y autorizar solicitudes entre diferentes partes en un sistema o aplicación web.
- **SQL:** es un lenguaje de programación utilizado para gestionar y manipular bases de datos relacionales. Es un estándar reconocido internacionalmente para interactuar con sistemas de gestión de bases de datos y permite realizar diversas operaciones, como consultar, insertar, actualizar y eliminar datos de una base de datos.
- **HTTP:** es un protocolo de comunicación utilizado para el intercambio de datos entre un cliente y un servidor. En este proyecto, para comunicar el cliente con la API.
- **Framework:** Un framework es un conjunto de herramientas, bibliotecas, componentes y estándares que proporcionan una estructura y una base de trabajo común para el desarrollo de software. Proporciona una infraestructura predefinida y reutilizable que permite a los desarrolladores construir aplicaciones de manera más eficiente y con menos esfuerzo. En este proyecto se usa el framework .NET de Microsoft.
- **LINQ:** es una característica del framework de .NET que permite realizar consultas y manipulación de datos de manera integrada en el lenguaje de programación. Proporciona una forma uniforme de consultar y transformar datos de diferentes fuentes, como colecciones, bases de datos y servicios web, utilizando una sintaxis similar a SQL.

- **Middleware:** un middleware es un software que actúa como intermediario entre distintos sistemas, aplicaciones o componentes. En este proyecto se trata de un middleware de seguridad encargado de la autenticación de usuarios.