



FACULTAD DE CIENCIAS

# Memoria de Prácticas Externas

*Inglis*

Memoria de prácticas externas realizadas en

Evenbytes S.L

**Alumno:** Alberto Campuzano Solórzano  
Grado en Matemáticas

**Periodo:** 16/06/2025 – 29/08/2025

**Tipo de prácticas:** Curriculares + Extracurriculares

**Número de horas:** 300 + 50

**Tutor externo:** Jacinto Pelayo, CEO  
**Tutor académico:** Alicia Nieto Reyes

Santander, 28 de julio de 2025

---

## Resumen

Durante mis prácticas en Evenbytes S.L., una consultora tecnológica especializada en soluciones cloud y transformación digital, he desarrollado competencias en el diseño de aplicaciones web y, de manera principal, en ciencia de datos. En la fase inicial, centrada en el desarrollo web, trabajé con el *framework* Angular, el entorno de Google Cloud Platform (GCP) y el sistema de control de versiones *GitHub*. Posteriormente, participé en diversos proyectos, destacando especialmente uno orientado a la implementación de un modelo de predicción del precio de la electricidad y a la optimización de los procesos de fundición de unos hornos para la reducción del consumo energético. Esta experiencia me ha permitido integrar los conocimientos adquiridos en los grados de Matemáticas y Física, además de proporcionarme una visión más clara y aplicada del entorno profesional tecnológico.

**Palabras clave:** Ciencia de datos, Optimización, Modelos de predicción, Python, GPC.

## Abstract

During my internship at Evenbytes S.L., a technology consulting firm specializing in cloud solutions and digital transformation, I developed skills in web application design and, primarily, in data science. In the initial phase, focused on web development, I worked with the Angular framework, the Google Cloud Platform (GCP) environment, and the GitHub version control system. Later, I participated in various projects, particularly one aimed at implementing a model for predicting electricity prices and optimizing the smelting processes of furnaces to reduce energy consumption. This experience has allowed me to integrate knowledge acquired in Mathematics and Physics degrees, as well as providing me with a clearer and more applied view of the technological professional environment.

**Keywords:** Data science, Optimization, Predictive models, Python, GPC.

# Índice general

Siglas y acrónimos	5
Agradecimientos	6
<b>1 Introducción</b>	<b>7</b>
1.1 Elección	7
1.2 Objetivos	7
1.3 Centro	8
1.4 Práctica Externa	8
<b>2 Descripción de las prácticas</b>	<b>9</b>
2.1 Actividades desempeñadas	9
2.2 Formación	9
2.2.1 Angular	9
2.2.2 GCP	13
2.2.3 GitHub	14
2.3 Proyecto principal	15
2.3.1 Predicción de precios de la luz	15
2.3.2 Modelo físico	17
2.3.3 Optimización de la superficie de enfriamiento	19
2.4 Mención a otros proyectos	19
2.4.1 Traducciones	19
2.4.2 Distribución de personal	19
2.5 Relación entre formación recibida y actividades	19
2.6 Atención y asesoramiento recibido	20
<b>3 Conclusiones</b>	<b>21</b>
3.1 Data	21
3.1.1 ARIMA	21
3.1.2 Time series forecast	21
3.1.3 Neural Networks	21
.1 Codigos completo	21

# Índice de figuras

2.1	En este se pueden agregar y disminuir <i>jugadores</i> del entorno. . . . .	10
2.2	Formulario que permite variar los datos de una galaxia restaurarlos por los iniciales. . . . .	10
2.3	En este entorno se permite agregar elementos a la lista de manera general o condicionada. . . . .	11
2.4	Formulario similar a la anterior pero en este caso permitía operar con los datos mostrándolos transformados. . . . .	11
2.5	Segunda aplicación web con menú interactivo. . . . .	12
2.6	Fragmentos código entorno GPC. . . . .	13
2.7	Imagen GitHub . . . . .	14
2.8	Serie temporal del precio horario de la electricidad (mercado SPOT español). . . . .	16
2.9	Diagramas de autocorrelación (ACF) y autocorrelación parcial (PACF). . . . .	17
2.10	Residuos del modelo SARIMAX tras ajuste. . . . .	17
2.11	Evaluación de la varianza residual del modelo SARIMAX. . . . .	17

# Índice de cuadros

# Siglas y acrónimos

UC Universidad de Cantabria

GCP Google Cloud Platform (Plataforma Google Cloud)

TSF Time series forecasting (Predicción de series temporales)

TFT Time series forecasting transformer (Transformador de predicción de series temporales)

SARIMAX (Seasonal AutoRegressive Integrated Moving Average with eXogenous regressors) (Medias variables integradas autorregresivas estacionales con regresores exógenos)

CEO Chief Executive Officer (Director ejecutivo)

CTO Chief Technology Officer (Director tecnológico)

COO Chief Operating Officer (Director de operaciones)

URL Uniform Resource Locator (Localizador Uniforme de Recursos)

gif Graphics Interchange Format (Formato de Intercambio de Gráficos) <sup>1</sup>

CSV Comma separated values (valores separados por comas).

IA Inteligencia artificial

ACF Autocorrelation function (función de autocorrelación)

PACF Partial autocorrelation function (función de autocorrelación parcial)

API Application Programming Interface (Interfaz de Programación de Aplicaciones)

ESIOS Estructura de Intercambio de Información del Operador del Sistema (Operador del Sistema Eléctrico Español)

---

<sup>1</sup>Según la RAE, aunque GIF sea una sigla, la RAE lo considera ya un acrónimo lexicalizado (una palabra a todos los efectos), con minúsculas: «gif», «gifs» (en cursiva, por tratarse de una voz no adaptada, ya que en español a esa grafía le correspondería la pronunciación [jíf]) [1].

# Agradecimientos

[Texto opcional de agradecimientos.]

# Capítulo 1

## Introducción

### 1.1. Elección

El proceso que llevó a la decisión de incorporarme a Evenbytes estuvo influenciado por diversos factores. En primer lugar, tuve la fortuna de que amigos míos, ya graduados de esta misma facultad, realizasen las prácticas en esta empresa, teniendo estos además diferentes perfiles, de manera que contaba de inicio con la ventaja de conocer la empresa de manera previa. Además, durante la feria de empleo realizada el 30 de abril de este mismo año pude asistir a los diferentes puestos que situaban las empresas asistentes para promocionarse y captar posibles futuros trabajadores. Dentro de estos grupos, Evenbytes fue la empresa que más llamó mi atención gracias a un proyecto que involucraba la implementación de predicciones estadística, modelado físico y optimización matemática. Fue por tanto la conjunción de un proyecto que aunase las dos carreras en las que me estoy formando, con el conocimiento previo del buen ambiente de trabajo, la que hizo que me decantase por Evenbytes.

### 1.2. Objetivos

La decisión acerca del futuro laboral es una tarea complicada, y más aún cuando tu formación te otorga una plenitud de opciones. Además, me encontraba ante la indecisión de si continuar por el mundo académico o desligarme de lo *teórico* e insertarme en la aplicación de los conocimientos adquiridos. Ante esta situación, veía más necesario que tener un contacto directo con el mundo profesional para poder dilucidar cuál quería que fuera mi futuro. Entre las ramas de mi interés se encontraban principalmente el sector financiero y el tecnológico, ya que considero que ambas áreas ofrecen un campo de desarrollo muy amplio para los perfiles con formación en Física y Matemáticas.

Por tanto, buscar una empresa que encajase con estas características era indispensable. La mayor parte de las ofertas actuales, especialmente en ámbitos relacionados con el análisis de datos, la modelización estadística y el desarrollo de soluciones tecnológicas, se alinean con esta línea de interés. A la hora de seleccionar un centro que me aceptase, una de mis prioridades fue también el ambiente laboral, puesto que, en mi opinión, uno aprende más cuando se encuentra cómodo.

En este aspecto, Evenbytes me acogió de manera total, siendo un claro ejemplo de un entorno de trabajo saludable en el que se valora la iniciativa, la capacidad de adaptación y la aportación personal al proyecto común. Considero que realizar las prácticas en una empresa con estas características me permitía no sólo adquirir experiencia práctica sino también contrastar en primera persona cómo se aplican competencias transversales como el pensamiento crítico, toma de decisiones en proyec-



tos reales y la proactividad. Asimismo, esta oportunidad suponía un primer paso importante para explorar con mayor claridad qué caminos profesionales me gustaría recorrer en el futuro.

### 1.3. Centro

Evenbytes es una consultora tecnológica que impulsa la transformación digital de empresas mediante soluciones tecnológicas avanzadas. Con un recorrido de más de 10 años de experiencia, su objetivo es ayudar a las organizaciones a ser más eficientes, competitivas e innovadoras, adaptándose siempre a sus necesidades específicas.

Especializados en el desarrollo de software, infraestructuras en la nube y consultoría tecnológica, en Evenbytes se posiciona como un aliado estratégico para empresas de todos los tamaños, desde startups hasta grandes corporaciones. Son Google Cloud Partners y cuentan con experiencia consolidada en áreas como IA, Big Data e IoT, poniendo estas tecnologías al alcance de sus clientes para que puedan afrontar los retos de un entorno empresarial en constante evolución. Sus principales áreas de especialización incluyen el desarrollo de software personalizado, gestión de infraestructuras en la nube, automatización y optimización de procesos, Inteligencia artificial y Big Data e integración de sistemas y plataformas.

La empresa actualmente cuenta con 16 trabajadores y 4 estudiantes de prácticas. Entre los perfiles de los compañeros se encontraban principalmente el Grado universitario en Ingeniería Informática, Ingeniería de Telecomunicaciones, Grados superiores relacionados con la informática y y Grado en Matemáticas. La actividad está dividida en áreas de gerencia, ventas, consultoría, desarrollo y data science. Los 3 socios y principales responsables ocupan los cargos de CEO, CTO y COO

### 1.4. Práctica Externa

Las labores realizadas durante el tiempo de prácticas han sido diversas. Como bien es sabido, la formación tanto en matemáticas como en física dotan de una gran capacidad de polivalencia a la persona que la posee, por lo que los trabajos que he realizado han sido diversos. Previamente se ha mencionado que Evenbytes es una empresa que se encuentra en plena expansión dentro del sector de la transformación digital. Por ello, en las semanas iniciales de las prácticas llevé a cabo labores de formación relacionadas principalmente con esta rama: Programación en Angular, diseño de páginas web, entorno GCP, GitHub, etc. Tras haber realizado dichas tareas me incorporé en el equipo de Data Science. En este departamento fui incluido en varios proyectos que se describirán a continuación, en los que he aprendido diversos programas relacionados con la ciencia de datos como *BigQuery*, *Jupyter*, *Python* ... además de técnicas para el tratado de datos y predicciones como *ARIMA*, *SARIMAX* o *redes neuronales*. Adicionalmente, debido a las características de uno de los proyectos me vi involucrado en el modelado físico de un proceso termodinámico, por lo que también he puesto en práctica los conocimientos adquiridos en el grado en Física.

# Capítulo 2

## Descripción de las prácticas

### 2.1. Actividades desempeñadas

Durante mi periodo de prácticas me he visto involucrado en diversos proyectos y dado que las actividades desempeñadas han sido variadas, no es del todo posible crear un *log frame* que las recoja todas de manera organizada en el sentido temporal. Además, dichos proyectos no se han iniciado y finalizado durante mi estancia por lo que no se puede definir de una manera feaciente las fases en las que se encontraban. Por tanto, se diseccionará la descripción de las prácticas en dos partes. En primer lugar, una asociada a la formación, en la que se explicarán los conceptos básicos aprendidos en las primeras semanas de trabajo. A continuación, se expondrá el proyecto al que mayor tiempo dediqué por motivos de extensión, si bien incluirá una sección te menciones a estos otros trabajos.

### 2.2. Formación

Mis primeros pasos en el aprendizaje de las herramientas de trabajo de Evenbytes está dominado por: *Angular*, *entorno GCP* y *GitHub*. Estas tres herramientas forman el pilar básico de la empresa. Angular es el framework principal utilizado para el desarrollo de aplicaciones web. Como *partners* de Google, el entorno GCP proporciona la infraestructura en la nube preferida por la empresa y GitHub que es un sistema de control de versiones ampliamente extendido.

#### 2.2.1. Angular

Angular es un framework de desarrollo basado en *JavaScript* y *HTML*<sup>1</sup> para la creación de aplicaciones web dinámicas, que permite estructurar el código de manera modular y facilita la generación de interfaces interactivas. Naturalmente, una empresa dedicada en parte a la transformación digital requiere de personal capaz de implementar páginas web o aplicaciones para que los clientes hagan uso de ellas. Como ejemplo, uno de los productos de Evenbytes son los dispositivos de fichaje horario, por lo que los datos que se registran en el momento en el que un trabajador entra a su jornada laboral, además de ser tratados, se mostrarán para que la empresa empleadora pueda manejarlos, filtrarlos, etc.

El periodo dedicado al aprendizaje de este ítem fue inferior a una semana pero aun así el poder conocer el trabajo que supone la creación e implementación de páginas y aplicaciones fue satisfactorio y pude desarrollar ciertos ejemplos, como los formularios que se muestran a continuación:

---

<sup>1</sup>Técnicamente *TypeScript* pero esencialmente es lo mismo.

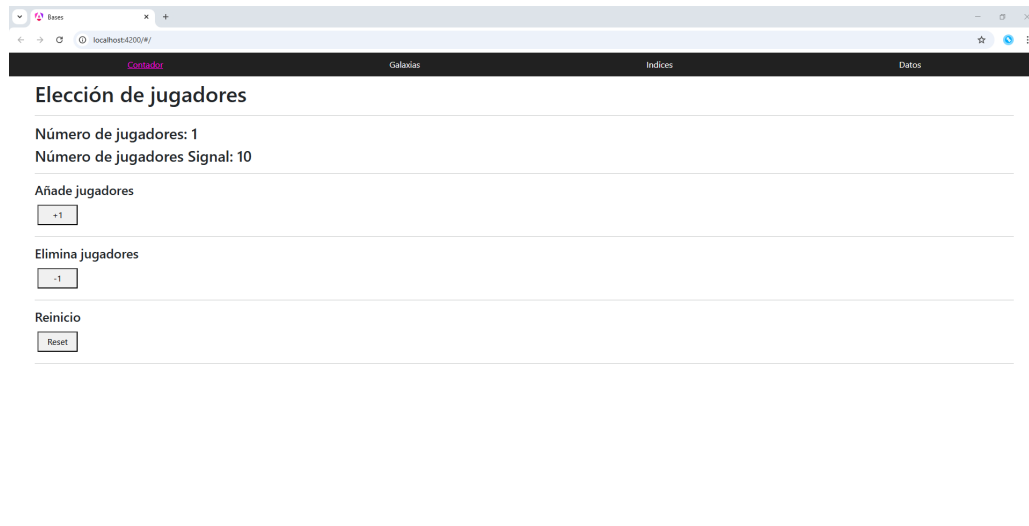


Figura 2.1: En este se pueden agregar y disminuir *jugadores* del entorno.

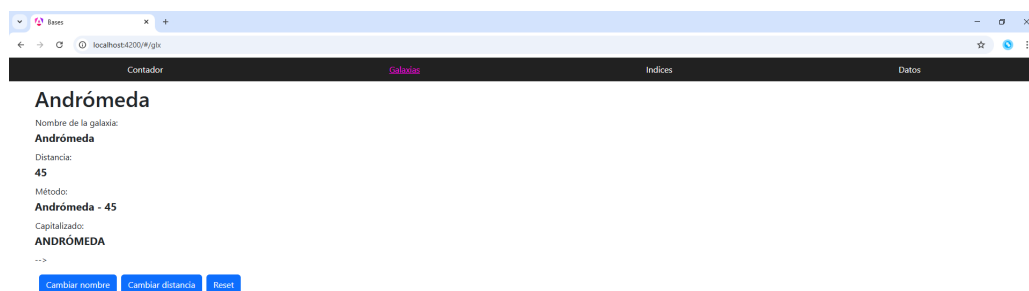


Figura 2.2: Formulario que permite variar los datos de una galaxia restaurarlos por los iniciales.

Figura 2.3: En este entorno se permite agregar elementos a la lista de manera general o condicionada.

Figura 2.4: Formulario similar a la anterior pero en este caso permitía operar con los datos mostrándolos transformados.

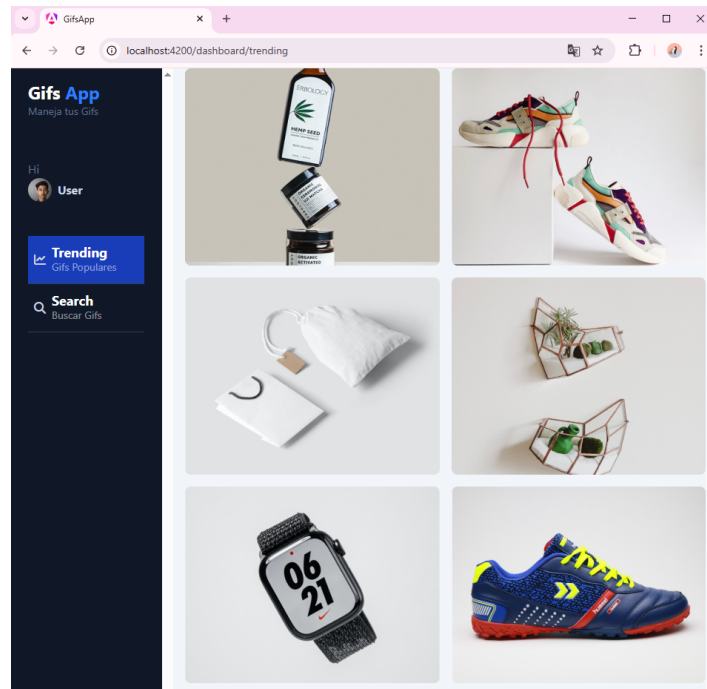
En las figuras<sup>2</sup> se muestran capturas de la primera página desarrollada, en la que se tenían diversos formularios, con sus *URLs* específicas. Los datos, inventados naturalmente, hacían referencia al nombre de galaxias, distancia, origen e identificador. La funcionalidad de la página era la creación de formularios cuyos datos se agregarán a listas de manera general, condicionada (en este caso particular por el origen o la distancia). Además, se añadió la posibilidad de exponer los datos, cambiarlos o restaurarlos.

En segundo lugar, se procedió a la creación de una aplicación más elaborada, que tuviera un menú interactivo como capa principal y que registrara en él ciertos datos como el usuario o una foto

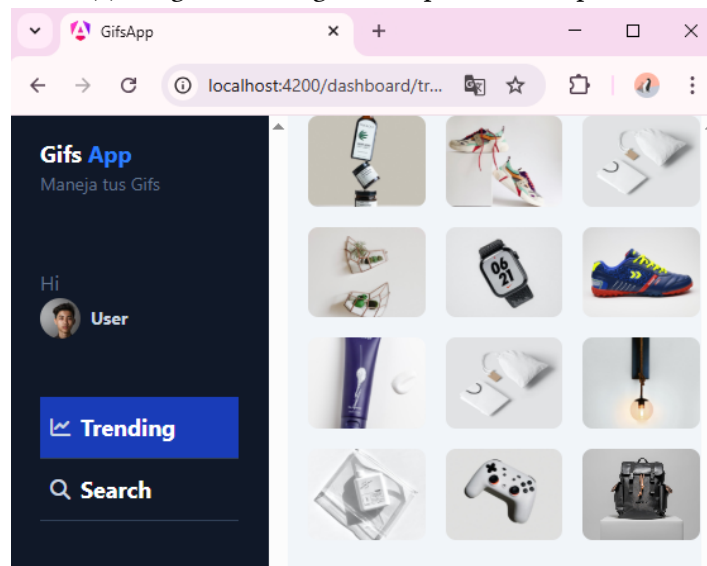
<sup>2</sup>La primera de ellas se corresponde con una prueba en la que se analizaba el comportamiento de distintas formas de definir variables en Angular y no tiene relación con las siguientes.

de perfil.

El navegador se comportaba de manera reactiva, adaptándose al tamaño de la pantalla, tal y como se muestra en la Figura 2.5. A la izquierda de las imágenes puede apreciarse el menú lateral, en el que aparecen los ítems (en este caso, dos) que permiten acceder al resto de modalidades de la página.



(a) Imagen de navegador en pantalla completa.



(b) Imagen de navegador en pantalla reducida.

Figura 2.5: Segunda aplicación web con menú interactivo.

El repositorio de *gifs* incorporado en la aplicación se organizaba mediante una pestaña denominada *trending*, que permitía ordenarlos. En este caso, como es evidente, no se podía disponer de un registro de las consultas de los usuarios, por lo que las entradas se organizaban de manera alfabética.

## 2.2.2. GCP

Este entorno proporciona la infraestructura en la nube necesaria para desplegar aplicaciones, almacenar datos y ejecutar servicios de manera flexible y segura. La empresa Evenbytes es, como ya se ha mencionado, *partner* de Google, por lo que brinda la posibilidad de trabajar con la mayor parte de funcionalidades de esta plataforma. Trabajar en la nube facilita el acceso compartido, la escalabilidad de los recursos y la automatización de procesos, lo que resulta especialmente valioso en proyectos donde intervienen varios miembros del equipo o que requieren alta disponibilidad.

En mi caso particular, implementé una *Cloud Function* encargada de realizar el proceso de extracción de información de una página web (llamado *web scraping*) y la generación de un fichero con los resultados. La *Cloud Function* permite ejecutar fragmentos de código en respuesta a eventos o invocaciones programadas, sin necesidad accionar explícitamente el programa. Desde Python realicé el código para conectarse a la página de Wikipedia sobre cocientes intelectuales por país, extraer las tablas relevantes y convertir los datos en un archivo CSV. El código de la función se desplegó en GCP mediante las herramientas de línea de comandos y quedó configurado para ejecutarse de forma automática a intervalos regulares.

Para esta última tarea hice uso de Cloud Scheduler, un servicio que permite programar tareas basadas en cron. La configuración del Scheduler se gestionó a través de Terraform. Gracias a esta combinación, cada ejecución de la función generaba automáticamente un nuevo bucket<sup>3</sup> de almacenamiento en Cloud Storage, en el que se guardaba el fichero CSV con los datos actualizados.

A continuación se muestran fragmentos de código de la programación del *web scraping*, la planificación en Terraform y el resultado del CSV.

```
13 def web_scraping():
14     # Busco en cada fila
15     filas = driver.find_elements(By.XPATH, '//*[@class="wikitable sortable jquery-tablesorter"]//tbody/tr')
16     print(f'Número de filas encontradas: {len(filas)}')
17     data = []
18
19     print("Comenzamos a buscar la información de las filas...")
20     # Las matrices
21     for i, fila in enumerate(filas):
22         try:
23             # Busco el elemento de la imagen
24             elemento_imagen = fila.find_element(By.XPATH, './img')
25             # Voy a coger ahora el nombre del país
26             texto_completo = elemento_imagen.get_attribute('alt')
27             nombre = texto_completo.replace("Bandera de ", "")
28             # Ahora extraigo los datos
29             td_completos = fila.find_elements(By.TAG_NAME, 'td')
30             # IQ en tercera columna por lo que lo imprimo directamente ahí
31             if len(td_completos) > 2:
32                 iq = td_completos[2].text
33                 position = td_completos[0].text
34                 data.append({
35                     'Position': position,
36                     'NombrePaís': nombre,
37                     'IQ': iq,
38                 })
39             else:
40                 print(f"La fila {i+1} no tiene suficientes columnas")
41         except Exception as error:
42             print(f"Error al procesar la fila {i+1}: {error}")
43
44     if driver:
45         driver.quit()
46         driver = None
47     print("Webdriver cerrado tras el scraping.")
48
49     # Creo el dataframe en pandas
50     df = pd.DataFrame(data)
51
52     if df.empty:
53         print("No se encontraron datos para exportar.")
54         # Aseguro: Respondo explícito para este caso
55         return "No se encontraron datos para exportar. El scraping pudo fallar.", 200
56
57     # Lo exporto a csv
58     csv_filename = f'iq_paises.csv'
59     csv_path = os.path.join('/tmp', csv_filename)
60     df.to_csv(csv_path, index=False, encoding='utf-8')
61     print(f"DataFrame guardado localmente en: {csv_path}")
62
```

(a) Fragmento del código de *web scraping* en Python [Listado 1].

```
1 provider "google" {
2   project = var.gcp_project_id
3   region  = var.gcp_region
4 }
5
6 # Obtener información del servicio de Cloud Run existente
7 data "google_cloud_run_service" "my_cloud_run_service" {
8   name     = var.cloud_run_service_name
9   location = var.gcp_region
10 }
11
12 # Crear el trabajo de Cloud Scheduler
13 resource "google_cloud_scheduler_job" "my_scheduler_job" {
14   name           = "invoke-iq-paises-alberto-daily"
15   description    = "Invoca el servicio de Cloud Run IQ Países diariamente"
16   schedule       = "0 9 * * 2,4"
17
18   http_target {
19     uri      = "https://iq-paises-998937693176.europe-west1.run.app"
20     http_method = "POST"
21
22     headers = {
23       "Content-Type" = "application/json"
24     }
25
26     oidc_token {
27       service_account_email = var.scheduler_service_account_email
28     }
29   }
30
31   time_zone = "Europe/Madrid"
32 }

```

(b) Ejemplo de configuración de Terraform para el *scheduler* [Listado 2].

Figura 2.6: Fragmentos código entorno GPC.

<sup>3</sup>Los buckets son contenedores que permiten almacenar objetos de manera segura y escalable. En este proyecto, cada bucket contenía los resultados de la extracción en formato CSV, de forma que quedaba un histórico de ejecuciones accesible a través de la consola de GCP o mediante las APIs correspondientes.

A modo de breve explicación de los códigos mostrados, para el caso del *web scraping* la idea principal es generar un navegador virtual al que se le manda a la página de la que queremos extraer los datos. De manera previa, se investiga el código HTML para saber la estructura de la página, de modo que creamos un bucle que busque los elementos que queremos y extraiga la información de ellos, es decir, si tenemos una estructura en la que todos los objetos tienen una propiedad llamada *nombre*, hacemos que nuestro código acceda a él para extraer todos los nombres de la página y guardarlos en un fichero.

Por otro lado, el scheduler es más sencillo, le introducimos los datos del proyecto, función a llamar... y definimos las órdenes bajo las que se ejecuta. En el código aparece en la línea 16 el fragmento `0 9 * * 2,4` que significa que se ejecute a las 9:00, *cualquier día* (numérico), *cualquier mes*, los *martes* y *jueves*.

### 2.2.3. GitHub

En último lugar estuvo el aprendizaje de GitHub, que actúa como sistema de control de versiones, permitiendo gestionar el historial de cambios en el código, coordinar el trabajo y asegurar que todas las contribuciones queden registradas de forma ordenada. Su uso es fundamental en un entorno profesional, ya que permite mantener la trazabilidad de los desarrollos, revisar modificaciones y garantizar la integridad de los proyectos a medida que evolucionan. Su estructura es muy sencilla y puede entenderse rápidamente pensando en grafos. Se crea una rama principal del proyecto y de ellas, cuando se quiere trabajar en una parte específica del mismo, salen hijas. De esta manera los cambios realizados se hacen fuera del proyecto principal, permitiendo guardar los cambios, avanzar únicamente en esa línea o abrir nuevas. Cuando una parte se da por revisada y acabada se unen ramas a la principal (padre), de manera que se va completando el trabajo. Los comandos básicos de Git son *commit*, que guarda los cambios realizados en el repositorio local; *pull*, que actualiza la copia local con las modificaciones del repositorio remoto; *push*, que sube los cambios locales al repositorio compartido; y *merge*, que combina diferentes ramas en una sola versión final.

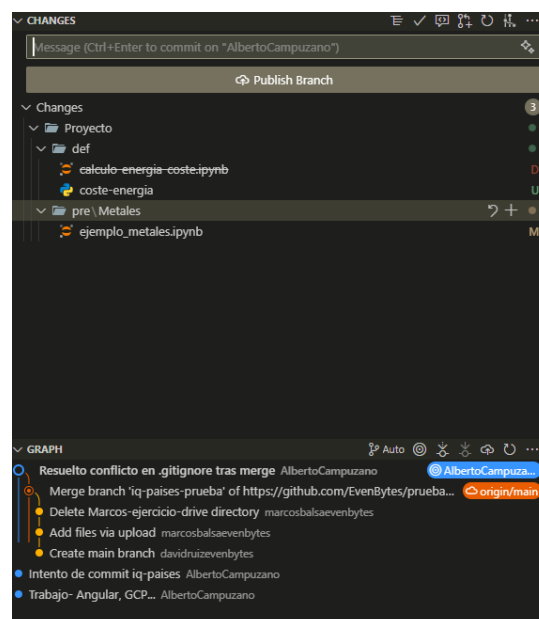


Figura 2.7: Imagen GitHub

*Nota: Por motivos de confidencialidad, ciertos contenidos no serán mostrados. Dado que el objetivo de estas prácticas es el manifiesto del conocimiento aprendido y técnicas implementadas no causará un gran perjuicio en la compresión de la misma.*

## 2.3. Proyecto principal

Este trabajo trataba de la optimización del gasto energético de una empresa que contaba con hornos de fundición. Para ello, se tuvieron que realizar predicciones del precio de la electricidad mediante modelos estadísticos. Posteriormente, se modelizó el proceso de fundición llevado a cabo en la fábrica y por último se trató de optimizar la selección de los materiales a fundir, dependiendo del espacio disponible en fábrica y el precio de la luz en cada tramo horario, estimado previamente.

A continuación, muestro una introducción a cada una de las partes mencionadas, con ejemplos o explicaciones del proceso. Como se ha dicho anteriormente, por motivos de confidencialidad no se pueden mostrar resultados o código explícito porque podría involucrar datos privados del proyecto o de la empresa cliente.

### 2.3.1. Predicción de precios de la luz

Para llevar a cabo la predicción del coste energético, se procedió en primer lugar al análisis de la serie temporal de precios de la electricidad. Este incluyó la generación de diversas gráficas, tales como series de tiempo o diagramas de autocorrelación simple (ACF) y parcial (PACF), con el fin de comprender la estacionalidad y el comportamiento de la varianza.

Concretamente, se utilizó como fuente de datos el mercado diario SPOT de electricidad en España. Los datos se obtuvieron a través de la API pública proporcionada por ESIOS (Sistema de Información del Operador del Sistema Eléctrico), que permite acceder a los precios horarios históricos, así como a otros indicadores del sistema eléctrico.

#### 2.3.1.1. Estudio y comprensión del mercado

Antes de abordar la predicción, se realizó un estudio del funcionamiento del mercado eléctrico español y de la normativa que afecta a las industrias electrointensivas. Esta revisión fue clave para entender qué variables influyen realmente en la factura energética de un cliente industrial, más allá del precio medio horario.

Por ejemplo, el cliente indicó que una parte importante del coste final se debía a excesos en el consumo en ciertas franjas horarias o a penalizaciones asociadas a picos de demanda. Esto motivó no solo la necesidad de predecir el precio por hora, sino también de incorporar restricciones operativas en la fase de optimización basadas en el conocimiento regulatorio del sector.

#### 2.3.1.2. Obtención de los datos

Los datos horarios del precio de la electricidad se descargaron mediante la API de ESIOS, para un rango histórico de 2023-2025 con el fin de capturar tanto patrones estacionales diarios como semanales y mensuales. Además, se recopilaban otras variables externas relevantes para la predicción, como el precio del gas natural, la demanda eléctrica real en España, y la generación horaria de energía solar y eólica. Estas variables exógenas se incorporaron al modelo con el objetivo de mejorar la



capacidad predictiva, al reflejar condiciones del sistema que tienen un impacto directo en la formación del precio del mercado. Adicionalmente, se implementó un proceso automático de actualización diaria de los datos, de modo que su pudiera reentrenar el modelo de manera regular.

### 2.3.1.3. Elección del modelo

Se trabajó con dos enfoques distintos para la predicción: un modelo estadístico basado en SARI-MAX y un modelo de *deep learning* basado en *Temporal Fusion Transformers* (TFT).

El modelo *SARIMAX* permite modelar series temporales con estacionalidad y con variables exógenas. En este caso, se consideró una estacionalidad diaria de 24 horas, con órdenes de diferenciación  $d = D = 1$ , y se dejaron abiertos los parámetros  $p, q, P, Q$  a seleccionar según el análisis de los correlogramas ACF y PACF. Las variables exógenas incluyeron la demanda real, la producción eólica y solar, y el precio del gas. La selección del modelo se basó en el análisis de estacionariedad (test de Dickey-Fuller aumentado) e inspección visual de autocorrelaciones.

Por otro lado, se aplicó también un modelo *Temporal Fusion Transformers* (TFT), un tipo de red neuronal recurrente avanzada, diseñada específicamente para problemas de series temporales multivariadas. Este modelo combina mecanismos de atención con una arquitectura interpretable y es capaz de capturar relaciones complejas a largo y corto plazo. El TFT permite incorporar múltiples variables exógenas, manejar datos faltantes y adaptarse bien a estacionalidades no lineales. En este caso, se le proporcionó como entrada no solo la serie de precios, sino también las variables exógenas mencionadas anteriormente. El modelo fue entrenado para predecir el precio horario con horizonte de un mes y se validó mediante un conjunto de datos independiente (*test set*). Las figuras correspondientes al entrenamiento y validación se muestran a continuación.

### 2.3.1.4. Resultados

Ambos modelos permitieron generar una predicción horaria de los precios de la electricidad, que posteriormente se utilizó como entrada para el problema de optimización descrito en la sección siguiente. Las predicciones del modelo *SARIMAX* mostraron una buena capacidad para capturar patrones estacionales regulares, mientras que el modelo TFT demostró una mejor adaptación a comportamientos más complejos o inusuales del sistema eléctrico.

El uso combinado de modelos permitió contrastar la robustez de las predicciones, y facilitó el diseño de estrategias operativas más informadas para la gestión energética en contextos industriales.

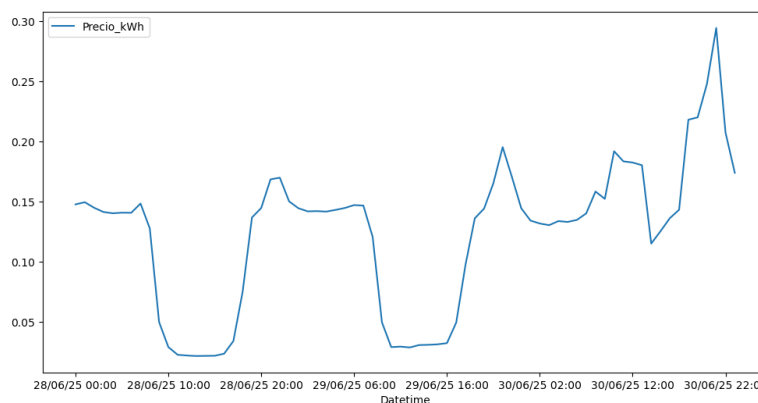


Figura 2.8: Serie temporal del precio horario de la electricidad (mercado SPOT español).

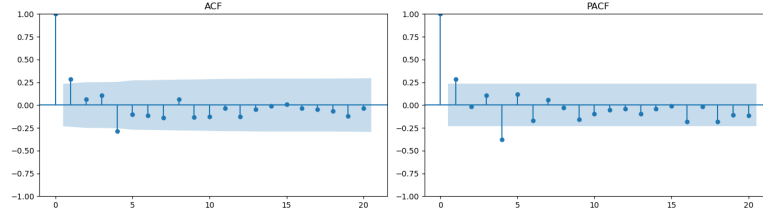


Figura 2.9: Diagramas de autocorrelación (ACF) y autocorrelación parcial (PACF).

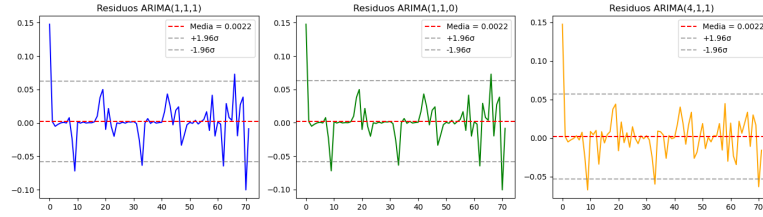


Figura 2.10: Residuos del modelo SARIMAX tras ajuste.

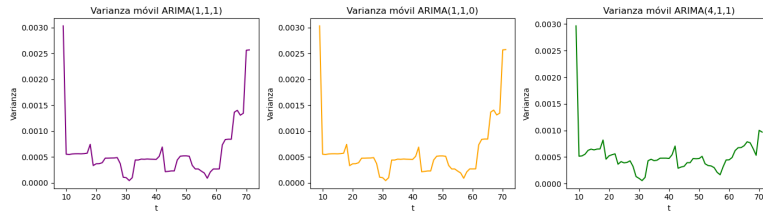


Figura 2.11: Evaluación de la varianza residual del modelo SARIMAX.

## 2.3.2. Modelo físico

### 2.3.2.1. Situación simplificada: Primera aproximación

Dado un metal conocido (X) de temperatura de fusión  $T_f$ , calor específico en los estados sólido y líquido,  $c_{sol}$  y  $c_{liq}$  respectivamente, entonces a la hora de conocer cuánta energía será necesaria tendremos que tener en cuenta:

- I El metal X a una temperatura inicial  $T_0$  debe ser calentado hasta la temperatura de fusión  $T_f$ , esto dependerá del *calor específico* del material en el estado *sólido*, que viene a ser el *coste* energético para aumentar la temperatura del mismo.
- II Una vez alcanzado dicho objetivo el material cambia de estado de la materia requiriendo una nueva cantidad de calor dependiente del *calor latente* del material, denominado como  $L_{s-1}$ .
- III Finalmente supondremos que seguiremos elevando la temperatura por lo que, de manera análoga al paso I, tendremos que tener en cuenta esta energía. Llamaremos a este límite  $T_c$

Para realizar dichos cálculos, haremos uso de las fórmulas de calor para calor específico y transiciones de fase.

$$Q_{pf} = m \cdot c_{sol} \cdot (T_f - T_0). \quad (2.1)$$

$$Q_f = m \cdot L. \quad (2.2)$$

$$Q_e = m \cdot c_{liq} \cdot (T_c - T_f). \quad (2.3)$$

De este modo,  $Q = Q_{pf} + Q_f + Q_e$ . Finalmente, deberíamos tener en cuenta que los hornos no son ideales y tenemos una cierta pérdida de energía. Este hecho, cuantificado por el rendimiento  $\eta \in [0, 1]$ , provoca un aumento en el calor total necesario:  $Q^{\text{tot}} = Q/\eta$ .

### 2.3.2.2. Situación general

Nos situamos ahora ante el problema general, en el que partimos de temperaturas iniciales cercanas a la ambiente ( $T_0 \simeq 25^\circ\text{C}$ ) mientras que llegamos por lo menos a  $T = 1450^\circ\text{C}$ . En este caso tendremos que aplicar la formulación general:

$$Q = m \cdot \int_{T_a}^{T_b} c(T) dT, \quad (2.4)$$

donde  $m$  es la masa del compuesto X y  $c(T)$  es la función del calor específico con la temperatura. Para poder operar con esta expresión sería necesario consultar datos tabulados sobre el material y una vez obtenidos los datos de  $c$  para distintas temperaturas interpolamos obtener una función con la que calcular, de manera aproximada, el calor en la Ecuación (2.4). De este modo llegaríamos a una situación similar a la descrita en la Apartado 2.3.2.1, obteniendo  $Q^{\text{tot}}$ . En ambos casos, como se va a tener desarrollo polinómico de  $c(T)$ , podemos calcular

$$Q = m \cdot \int_{T_a}^{T_b} c(T) dT = m \cdot \int_{T_a}^{T_b} [a_0 + a_1 T + \dots + a_n T^n] dT = \quad (2.5)$$

$$= m \cdot \left[ a_0 T + \frac{a_1}{2} T^2 + \dots + \frac{a_n}{n+1} T^{n+1} \right] \Big|_{T_a}^{T_b} = m \cdot p(T_a, T_b), \quad (2.6)$$

donde sustituyendo obtenemos el valor deseado. Para el primer caso, tenemos que  $T_a$  coincide con la temperatura *ambiente* de la pieza mientras que  $T_b$  con la de fusión del material X. Por otro lado, para la etapa final, la temperatura de fusión coincide con la inicial y la final, supondremos que es  $1450^\circ\text{C}$  para todas las piezas.

### 2.3.2.3. Traducción matemática

La situación anterior debe ser convenientemente planteada como restricciones, de cara a la resolución del problema de optimización con el que nos encontramos. Aunando lo descrito en la Apartado 2.3.2.2 podemos escribir que el calor  $Q$  necesario para la pieza X es:

$$Q = Q_{pf} + Q_f + Q_e + Q' = m \cdot (p_I + L + p_{III}) + Q', \quad (2.7)$$

Donde el término  $Q'$  es la energía debida a un requerimiento del fabricante, tanto de precalentamiento como de pos-fundición.

Finalmente, debería considerarse el rendimiento  $\eta$  de los hornos. Supondremos, salvo que se indique lo contrario, que ambos hornos presentan el mismo rendimiento. Adicionalmente, dado que este valor es una constante, tenemos que  $\min \eta f(x) = \eta \min f(x)$  por lo que prescindiremos de este valor puesto que no influye en la función objetivo y únicamente debería ser considerado en última instancia, a la hora de predecir el gasto estimado. Ya por último, se ha obtenido el calor  $Q$  con unidad (J) *julios*, mientras que habitualmente el coste eléctrico se da en €/kWh por lo que deberemos realizar la debida conversión. Esta, al igual que pasa con el rendimiento, influye en el valor numérico final pero no en el problema a optimizar.

### 2.3.3. Optimización de la superficie de enfriamiento

El presente problema aborda la distribución de las piezas metálicas recién fundidas sobre la superficie de enfriamiento disponible en la nave. Claramente, el objetivo es maximizar el número total de piezas que pueden colocarse simultáneamente. Esta superficie cuenta además con restricciones físicas relacionadas con la seguridad y operatividad del proceso, como márgenes laterales y pasillos interiores obligatorios de al menos 1.5 metros de ancho.

Cada pieza fundida es clasificado según su diametro y para su manipulación ha sido previamente colocada en una caja cuya dimensión se determina en función del tipo correspondiente. Estas cajas se disponen en planta sobre la superficie, sin posibilidad de apilarse. Una vez colocadas, las cajas no pueden solaparse ni invadir los pasillos o márgenes de seguridad. Es decir, deben respetarse las distancias mínimas entre ellas y con respecto a los bordes del área de enfriamiento. Esto impone restricciones geométricas estrictas sobre su posicionamiento.

El objetivo final es obtener una disposición óptima que maximice la cantidad total de piezas colocadas, permitiendo una planificación eficiente del enfriado. Dado el carácter discreto y combinatorio del problema, este puede abordarse mediante algoritmos de optimización específicos como métodos de empaquetado 2D, algoritmos genéticos o formulaciones de programación entera si se requiere una solución exacta.

## 2.4. Mención a otros proyectos

En esta sección se nombran y explican brevemente otros proyectos en los que he participado, en orden ascendente de horas invertidas.

### 2.4.1. Traducciones

Este proyecto trataba del desarrollo de una aplicación para acceder a manuales de rescate de vehículos, en la cual era necesaria, de manera previa, traducir una gran cantidad de documentos técnicos. Estos manuales contienen información crítica para los equipos de emergencia, por lo que era esencial disponer de versiones traducidas y clasificadas adecuadamente para facilitar su consulta en situaciones de intervención.

Mi participación estuvo enfocada en el desarrollo de herramientas automáticas para la obtención, traducción y organización de los manuales. Para ello, implementé código que permitía descargar los documentos desde distintas fuentes, procesarlos y traducirlos de forma automática utilizando la API de Gemini. Posteriormente, los archivos eran clasificados por marca, modelo y año.

### 2.4.2. Distribución de personal

INTRODUCIR WORD

## 2.5. Relación entre formación recibida y actividades

hola

---

## 2.6. Atención y asesoramiento recibido

hola

# Capítulo 3

## Conclusiones

[Reflexión personal, evaluación y recomendaciones.]

Incluyo a continuación informacion que pudiera ser relevante, completando a lo expuesto previamente que, por extensión, se ha omitido.

### .1. Codigos completo

```
1 from flask import Flask, request
2 from selenium import webdriver
3 from selenium.webdriver.common.by import By
4 import json
5 import os
6 import pandas as pd
7 from google.cloud import storage
8 from google.oauth2 import service_account
9
10 app = Flask(__name__)
11
12 @app.post("/")
13 def web_scraping():
14     """
15     Funcion que accede a la pagina de la wikipedia que contiene el
16     listado por paises de promedios de iq
17     extrae de ellas el pais e iq y crea un bucket con un csv con estos
18     datos
19     """
20     driver = None
21
22     try:
23         print("Cargamos las configuraciones del webdriver")
24         chrome_options = webdriver.ChromeOptions()
25         chrome_options.add_argument("--headless")
26         chrome_options.add_argument("--disable-gpu")
27         chrome_options.add_argument("window-size=1024,768")
28         chrome_options.add_argument("--no-sandbox")
29         chrome_options.add_argument("--disable-dev-shm-usage")
```

```

29 driver = webdriver.Chrome(options=chrome_options)
30 print("WebDriver inicializado con éxito.")
31
32 # Cargo la pagina en la que quiero scrappear
33 driver.get('https://es.wikipedia.org/wiki/
34           CI_y_la_riqueza_de_las_naciones')
35 print("Pagina cargada con éxito.")
36
37 # Busco en cada fila
38 filas = driver.find_elements(By.XPATH, '//*[@class = "
39           wikitable sortable jquery-tablesorter"]//tbody//tr')
40 print(f"Numero de filas encontradas: {len(filas)}")
41 data = []
42
43 print("Comenzamos a buscar la informacion de las filas...")
44 # Las recorro
45 for i, fila in enumerate(filas):
46     try:
47         # Tomo el elemento de la imagen
48         elemento_imagen = fila.find_element(By.XPATH, '//*[@img']')
49         # Voy a coger ahora el nombre del pais
50         texto_completo = elemento_imagen.get_attribute('alt')
51         nombre = texto_completo.replace("Bandera de ", "")
52         # Quiero extraer los iq ahora
53         td_completos = fila.find_elements(By.TAG_NAME, 'td')
54         # IQ en tercera columna por lo que lo printeo
55         directamente ahi
56         if len(td_completos) > 2:
57             iq = td_completos[2].text
58             posicion = td_completos[0].text
59             data.append({
60                 'Posicion': posicion,
61                 'NombrePais': nombre,
62                 'IQ': iq,
63             })
64         else:
65             print(f"La fila {i+1} no tiene suficientes columnas")
66     except Exception as error:
67         print(f"Error al procesar la fila {i+1}: {error}")
68
69 if driver:
70     driver.quit()
71     driver = None
72 print("WebDriver cerrado tras el scraping.")
73
74 # Creo el dataframe en pandas
75 df = pd.DataFrame(data)
76
77 if df.empty:
78     print("No se encontraron datos para exportar.")
79     # Asegurado: Retorno explicito para este caso

```

```

77         return "No se encontraron datos para exportar. El scraping
78             pudo fallar.", 200
79
80     # Lo exporto a csv
81     csv_filename = 'iq_paises.csv'
82     csv_path = os.path.join('/tmp', csv_filename)
83     df.to_csv(csv_path, index=False, encoding='utf-8')
84     print(f"DataFrame guardado localmente en: {csv_path}")
85
86     # Compruebo el tipo de dato
87     print(df.dtypes)
88
89     # Cloud storage
90     # Nombre del JSON
91     archivo_key = 'sandbox_estudiantes_key.json'
92     # Nombre del bucket
93     nombre_bucket = 'test_tablas_alberto'
94     # Nombre del archivo CSV en GCS (usamos el mismo nombre que el
95         local)
96     iq_paises_cgs = csv_filename
97     # Creo las credenciales
98     credenciales = service_account.Credentials.
99         from_service_account_file(archivo_key)
100     # Inicializo con las credenciales
101     storage_cliente = storage.Client(credentials=credenciales)
102     # Tomamos la referencia del bucket
103     bucket = storage_cliente.bucket(nombre_bucket)
104     # Sube el archivo al bucket de GCS
105     blob = bucket.blob(iq_paises_cgs)
106     blob.upload_from_filename(csv_path)
107     print(f"Archivo '{iq_paises_cgs}' subido exitosamente a 'gs://{
108         nombre_bucket}/{iq_paises_cgs}'")
109     # Elimino la ruta al csv
110     os.remove(csv_path)
111     print(f"Archivo local '{csv_path}' eliminado.")
112     return "Web scraping y subida a GCS completados exitosamente.", 2
113         00
114
115 except FileNotFoundError:
116     error_msg = f"Error: No se encontro el archivo de clave JSON '{
117         archivo_key}'. Asegurate de que esta en la raiz del proyecto
118         Docker."
119     print(error_msg)
120     return error_msg, 500
121
122 except Exception as error:
123     error_msg = f"Error inesperado durante la ejecucion: {error}"
124     print(error_msg)
125     return error_msg, 500
126
127 finally:
128     # Si driver != None lo cerramos, por si ha habido algun fallo y
129         no se cierra

```



```

120     if driver:
121         driver.quit()
122         print("WebDriver cerrado en el bloque finally final.")
123
124
125 if __name__ == "__main__":
126     app.run(host="0.0.0.0", port=8080, debug=True)

```

Listing 1: Código de web scrapping completo.

```

1 provider "google" {
2     project = var.gcp_project_id
3     region  = var.gcp_region
4 }
5
6 # Obtener informacion del servicio de Cloud Run existente
7 data "google_cloud_run_service" "my_cloud_run_service" {
8     name      = var.cloud_run_service_name
9     location  = var.gcp_region
10 }
11
12 # Crear el trabajo de Cloud Scheduler
13 resource "google_cloud_scheduler_job" "my_scheduler_job" {
14     name          = "invoke-iq-paises-alberto-daily"
15     description   = "Invoca el servicio de Cloud Run IQ paises diariamente"
16     schedule      = "0 9 * * 2,4"
17
18     http_target {
19         uri          = "https://iq-paises-998937693176.europe-west1.run.app"
20         http_method  = "POST"
21
22         headers = {
23             "Content-Type" = "application/json"
24         }
25
26         oidc_token {
27             service_account_email = var.scheduler_service_account_email
28         }
29     }
30
31     time_zone = "Europe/Madrid"
32 }

```

Listing 2: Código de Terraform para el Google Scheduler.

# Bibliografía

- [1] R. A. Española. «RAE informa.» Tweet. dirección: <https://x.com/RAEinforma/status/961534627046416384?lang=es>.