



**INSTITUTO POLITECNICO NACIONAL**  
**CENTRO DE INVESTIGACION EN COMPUTACIÓN**



## **Introducción a CHATGPT con Python**

### **PRÁCTICA 102**

**ALUMNO:**

**Félix Saucedo Garnica**

**INSTRUCTOR:**

**Alan Badillo Salas**

Ciudad de México. Agosto 2023

# **1. JUSTIFICACIÓN**

Las herramientas de inteligencia artificial y las tecnologías como ChatGPT están transformando la manera en que las personas aprendemos y aplicamos los conocimientos en diversos campos. La justificación para el uso de ChatGPT como herramienta de aprendizaje y para ser experimentado en su uso radica en las ventajas que ofrece para el acceso a información de manera prácticamente instantánea, la personalización del aprendizaje, y la exploración creativa. Además, aprender a utilizar ChatGPT también conlleva beneficios para el manejo de tecnologías, se mejora la comunicación, resolución de problemas y preparación para el futuro laboral. Si integramos ChatGPT en el proceso de aprendizaje beneficia sobremanera a los estudiantes para adquirir conocimientos de forma clara y también para desarrollar habilidades prácticas relevantes en un mundo cada vez más tecnológico.

## 2. INTRODUCCIÓN

ChatGPT es un modelo de lenguaje desarrollado por OpenAI, basado en la arquitectura GPT (Generative Pre-trained Transformer). GPT es un tipo de modelo de inteligencia artificial que utiliza el aprendizaje profundo para comprender y generar texto humano de manera coherente y contextual.

Específicamente, ChatGPT se entrena para generar respuestas en lenguaje natural en forma de conversaciones o diálogos. Esto significa que puede entender preguntas y comandos escritos por los usuarios y responder de manera coherente y relevante, creando la ilusión de una conversación con una entidad inteligente.

El entrenamiento de ChatGPT implica exponer al modelo a grandes cantidades de texto de conversaciones humanas, lo que le permite aprender patrones lingüísticos, estructuras gramaticales, y adquirir un conocimiento general sobre diferentes temas. Esto le permite generar respuestas que se ajustan al contexto de una conversación y que suenan como si fueran escritas por una persona.

El objetivo detrás de modelos como ChatGPT es proporcionar una forma eficiente de interactuar con sistemas de inteligencia artificial y aprovechar su capacidad para comprender y generar lenguaje humano. Puede ser utilizado en una variedad de aplicaciones, como asistentes virtuales, chatbots, generación de texto, respuestas a preguntas, entre otros. Sin embargo, es importante recordar que, si bien puede generar respuestas coherentes y útiles, el modelo no posee un verdadero entendimiento ni conciencia, sino que se basa en patrones aprendidos del texto con el que fue entrenado.

### **3. DESARROLLO**

En esta práctica, fusionaremos las capacidades de ChatGPT y Python para procesar un archivo CSV, generando informes estadísticos a partir de los datos proporcionados. Luego, exploraremos la extracción de texto contenido en una imagen, realizando un análisis posterior mediante ChatGPT. Para culminar, llevaremos a cabo un caso de estudio que pondrá en práctica nuestros conocimientos recién adquiridos, incluyendo la manipulación de imágenes de manera efectiva.

#### **3.1 REPORTE ESTADÍSTICO**

Una forma de comunicar los resultados de un estudio o análisis de datos de manera clara y comprensible para ayudar en la toma de decisiones es a través de un reporte estadístico. Estos informes pueden ser generados por investigadores, analistas, científicos de datos, economistas y profesionales de diversas disciplinas que trabajan con datos numéricos. Un reporte estadístico es un documento que presenta y analiza información cuantitativa (datos numéricos) utilizando métodos estadísticos.

La relevancia de los reportes estadísticos abarca una amplia variedad de esferas en la sociedad, dando lugar a su extenso uso en diversos campos, tales como las Ciencias Sociales, Negocios y Economía, Medicina y Salud, Educación, Investigación Científica, Ingeniería y Tecnología, entre otros. Estos informes desempeñan un papel fundamental al simplificar una serie de tareas cruciales, como la Toma de Decisiones, la Identificación de Tendencias, la Evaluación de Resultados, la Verificación de Hipótesis, y una diversidad de aplicaciones valiosas en la vida cotidiana.

En los siguientes ejercicios veremos como ChatGPT y Python nos pueden ayudar a obtener reportes estadísticos.

### 3.1.1 Abrir un archivo CSV desde una URL usando Pandas

Solicita a CHATGPT que genere el código en Python para abrir un archivo weight-height.csv usando Pandas desde la URL <https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv>

#### Solicitud:



Genera el código en Python usando Pandas para abrir un archivo weight-height.csv desde la URL:  
<https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv>

#### Resultado:



Por supuesto, aquí tienes el código en Python utilizando la biblioteca Pandas para abrir el archivo CSV desde la URL proporcionada:

```
import pandas as pd

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Cargar el archivo CSV en un DataFrame
data = pd.read_csv(url)

# Mostrar las primeras filas del DataFrame
print(data.head())
```

Este código descargará el archivo CSV desde la URL proporcionada y lo cargará en un DataFrame de Pandas. Luego, imprimirá las primeras filas del DataFrame para que puedas ver los datos.

#### Resultado de ejecutar el código obtenido:

	Gender	Height	Weight
0	Male	73.847017	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740856
3	Male	71.730978	220.042470
4	Male	69.881796	206.349801
..	...	...	...
95	Male	65.117485	165.717112
96	Male	71.701234	193.094164
97	Male	66.832878	180.683887

```
98    Male    66.471275    172.773723
99    Male    69.411526    177.470616

[100 rows x 3 columns]
```

### 3.1.2 Imprimir 10 valores aleatorios en Pandas

Solicita a CHATGPT que genere el código que imprima 10 valores aleatorios del dataset anterior usando la función `sample` de Pandas

Solicitud:



Genera el código en Python que imprima 10 valores aleatorios del dataset anterior usando la función `sample` de Pandas

Resultado:



Por supuesto, aquí tienes el código en Python para imprimir 10 valores aleatorios del conjunto de datos utilizando la función `sample` de Pandas:

```
valores_aleatorios = data.sample(10)

print(valores_aleatorios)
```

En este código, estamos utilizando la función `sample` de Pandas para obtener una muestra aleatoria de 10 filas del DataFrame cargado desde la URL. Luego, simplemente imprimimos esa muestra aleatoria utilizando `print`.

Resultado de ejecutar el código obtenido:

	Gender	Height	Weight
8068	Female	59.447758	108.111644
968	Male	64.922336	179.409092
7307	Female	67.022934	181.793069
7882	Female	60.458613	118.428035
9810	Female	60.025917	126.460830
8417	Female	67.320824	153.522260
8851	Female	61.930054	117.192630
1218	Male	69.475608	203.021282
3845	Male	68.676641	182.183981
655	Male	65.224692	160.836074

### 3.1.3 Obtención de valores únicos y sus conteos de la columna Gender

Solicita a CHATGPT que genere el código que describa la columna `Gender` que es categórica, pide que extraiga los valores únicos y sus conteos

Solicitud:



Genera el código que describa, de la información anterior, la columna `Gender` que es categórica, extrae los valores únicos y sus conteos

Resultado:



Por supuesto, aquí tienes el código para describir la columna "Gender" que es categórica, extrayendo los valores únicos y sus conteos:

```
descripcion_genero = data['Gender'].value_counts()

print("Descripción de la columna 'Gender':")
print(descripcion_genero)
```

En este código, estamos utilizando la función `value_counts()` de Pandas para contar los valores únicos en la columna "Gender" del DataFrame cargado desde la URL. Luego, imprimimos los valores únicos y sus conteos utilizando `print`. Esto te proporcionará una descripción de la distribución de géneros en el conjunto de datos.

Resultado de ejecutar el código obtenido:

```
Descripción de la columna 'Gender':
Male      5000
Female    5000
Name: Gender, dtype: int64
```

### 3.1.4 Obtención de estadísticos de la columna Height

Solicita a CHATGPT que genere el código que describa la columna `Height` que es numérica y obtenga los estadísticos: total, mínimo, máximo, promedio, desviación estándar, intervalo de confianza al 95%, y los cuartiles

## Solicitud:

CI

Genera el código en Python que, de la información anterior describa la columna `Height` que es numérica y obtenga los estadísticos: total, mínimo, máximo, promedio, desviación estándar, intervalo de confianza al 95%, y los cuartiles

## Resultado:



Por supuesto, aquí tienes el código en Python que describe la columna "`Height`" (Altura) del DataFrame cargado desde el archivo CSV y calcula los estadísticos solicitados:

```
import pandas as pd
import scipy.stats

url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"
data = pd.read_csv(url)

height_column = data['Height']

total_heights = len(height_column)
min_height = height_column.min()
max_height = height_column.max()
mean_height = height_column.mean()
std_deviation = height_column.std()
confidence_interval = scipy.stats.norm.interval(0.95,
loc=mean_height, scale=std_deviation/total_heights**0.5)
quartiles = height_column.quantile([0.25, 0.5, 0.75])

print("Total heights:", total_heights)
print("Minimum height:", min_height)
print("Maximum height:", max_height)
print("Mean height:", mean_height)
print("Standard deviation:", std_deviation)
print("Confidence interval (95%):", confidence_interval)
print("Quartiles:")
print("25th percentile (Q1):", quartiles[0.25])
print("50th percentile (median - Q2):", quartiles[0.5])
print("75th percentile (Q3):", quartiles[0.75])
```

Este código calcula los estadísticos solicitados para la columna "`Height`" del DataFrame, incluyendo el total de valores, el mínimo, el máximo, el promedio, la desviación estándar, el intervalo de confianza al 95% y los cuartiles (percentiles 25, 50 y 75). Ten en cuenta que para calcular el intervalo de confianza al 95%, se utiliza la distribución normal con un ajuste para el tamaño de la muestra.



Resultado de ejecutar el código obtenido:

```
Total heights: 10000
Minimum height: 54.2631333250971
Maximum height: 78.9987423463896
Mean height: 66.36755975482124
Standard deviation: 3.8475281207732293
Confidence interval (95%): (66.29214958935904, 66.44296992028345)
Quartiles:
25th percentile (Q1): 63.505620481218955
50th percentile (median - Q2): 66.31807008178464
75th percentile (Q3): 69.1742617268347
```

### 3.1.5 Obtención de estadísticos de la columna `Weight`

Solicita a CHATGPT que genere el código que describa la columna `Weight` que es numérica y obtenga los estadísticos: total, mínimo, máximo, promedio, desviación estándar, intervalo de confianza al 95%, y los cuartiles

Solicitud:



Genera el código Python que describa la columna `Weight`, de la información anterior, que es numérica y obtenga los estadísticos: total, mínimo, máximo, promedio, desviación estándar, intervalo de confianza al 95%, y los cuartiles

Resultado:



Por supuesto, aquí tienes el código en Python que describe la columna "`Weight`" (Peso) del DataFrame cargado desde el archivo CSV y calcula los estadísticos solicitados:

```
import pandas as pd
import scipy.stats

url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"
data = pd.read_csv(url)

weight_column = data['Weight']

total_weights = len(weight_column)
min_weight = weight_column.min()
max_weight = weight_column.max()
```

```

mean_weight = weight_column.mean()
std_deviation = weight_column.std()
confidence_interval = scipy.stats.norm.interval(0.95,
loc=mean_weight, scale=std_deviation/total_weights**0.5)
quartiles = weight_column.quantile([0.25, 0.5, 0.75])

print("Total weights:", total_weights)
print("Minimum weight:", min_weight)
print("Maximum weight:", max_weight)
print("Mean weight:", mean_weight)
print("Standard deviation:", std_deviation)
print("Confidence interval (95%):", confidence_interval)
print("Quartiles:")
print("25th percentile (Q1):", quartiles[0.25])
print("50th percentile (median - Q2):", quartiles[0.5])
print("75th percentile (Q3):", quartiles[0.75])

```

Este código realiza los mismos cálculos que el código anterior, pero esta vez se aplica a la columna "Weight" del DataFrame. Se calculan los estadísticos solicitados, incluyendo el total de valores, el mínimo, el máximo, el promedio, la desviación estándar, el intervalo de confianza al 95% y los cuartiles (percentiles 25, 50 y 75) para la columna de peso.

Resultado de ejecutar el código obtenido:

```

Total weights: 10000
Minimum weight: 64.700126712753
Maximum weight: 269.989698505106
Mean weight: 161.44035683283076
Standard deviation: 32.108439006519596
Confidence interval (95%): (160.81104299230498, 162.06967067335654)
Quartiles:
25th percentile (Q1): 135.8180513055015
50th percentile (median - Q2): 161.21292769948298
75th percentile (Q3): 187.16952486868348

```

### 3.1.6 Gráfica del Histograma de la columna Height

Solicita a CHATGPT que genere el código que grafique el histograma de la columna Height usando **Seaborn**

## Solicitud:

CI

Genera el código Python que grafique el histograma de la columna Height, de la información anterior, usando Seaborn

## Resultado:



Por supuesto, aquí tienes el código en Python que utiliza la biblioteca Seaborn para graficar un histograma de la columna "Height" del DataFrame:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"
data = pd.read_csv(url)

sns.set(style="whitegrid") # Configura el estilo de Seaborn

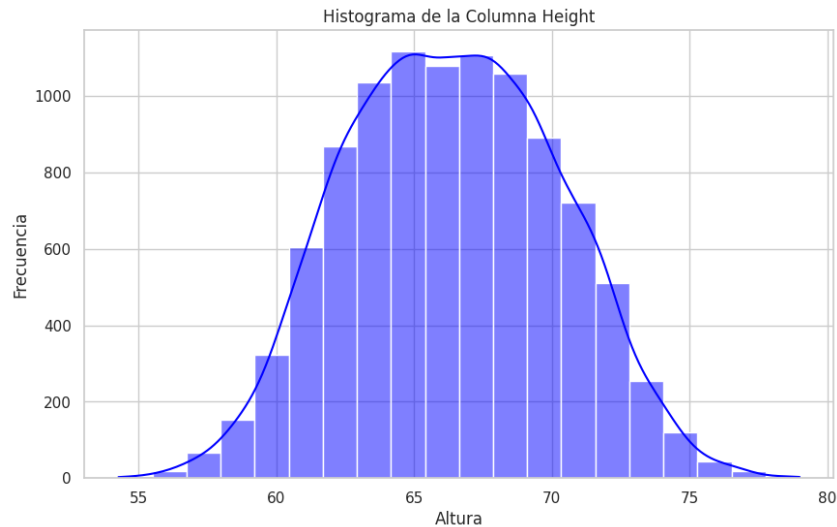
plt.figure(figsize=(10, 6)) # Tamaño de la figura

# Crea el histograma utilizando Seaborn
sns.histplot(data['Height'], bins=20, kde=True, color='blue')

plt.title('Histograma de la Columna Height')
plt.xlabel('Altura')
plt.ylabel('Frecuencia')
plt.show()
```

Este código utiliza Seaborn y Matplotlib para generar un histograma de la columna "Height". Se configura el estilo de Seaborn, se crea la figura con un tamaño específico y se utiliza la función `sns.histplot()` para dibujar el histograma. El parámetro `bins` controla la cantidad de barras en el histograma y el parámetro `kde` agrega una estimación de la densidad de kernel en la trama. Finalmente, se establecen los títulos y etiquetas adecuados para la figura antes de mostrarla con `plt.show()`. Asegúrate de tener instaladas las bibliotecas Seaborn y Matplotlib antes de ejecutar este código. Puedes instalarlas usando los comandos `pip install seaborn matplotlib`.

La grafica obtenida al ejecutar el código es:



### 3.1.7 Gráfica del Histograma de la columna Weight

Solicita a CHATGPT que genere el código que grafique el histograma de la columna `Weight` usando **Seaborn**

**Solicitud:**



Genera el código Python que grafique el histograma de la columna `Weight`, de la información anterior, usando Seaborn

**Resultado:**



Por supuesto, aquí tienes el código en Python que utiliza la biblioteca Seaborn para graficar un histograma de la columna "Weight" del DataFrame:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```

url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"
data = pd.read_csv(url)

sns.set(style="whitegrid") # Configura el estilo de Seaborn

plt.figure(figsize=(10, 6)) # Tamaño de la figura

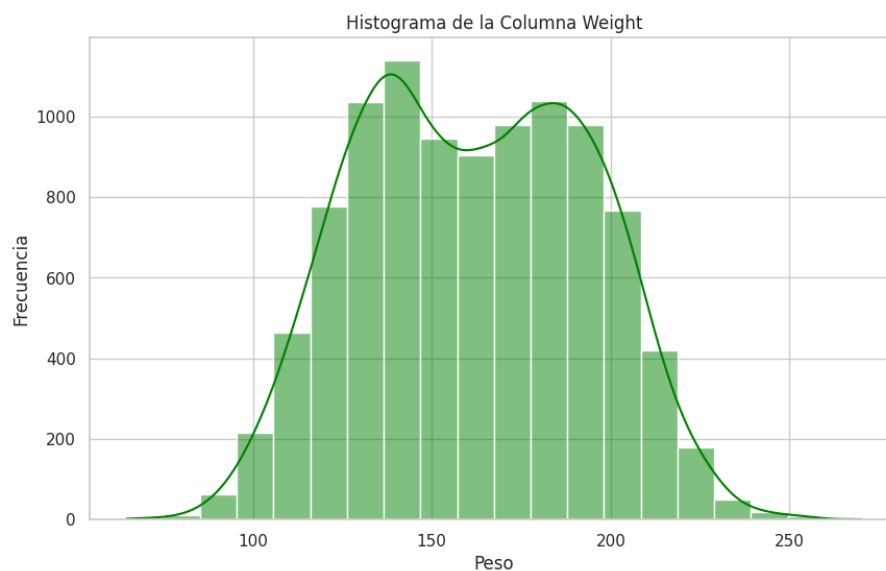
# Crea el histograma utilizando Seaborn
sns.histplot(data['Weight'], bins=20, kde=True, color='green')

plt.title('Histograma de la Columna Weight')
plt.xlabel('Peso')
plt.ylabel('Frecuencia')
plt.show()

```

Este código es similar al anterior, pero esta vez crea un histograma de la columna "Weight" en lugar de la columna "Height". Se configura el estilo de Seaborn, se establece el tamaño de la figura y se utiliza la función `sns.histplot()` para trazar el histograma. Los parámetros `bins` y `kde` controlan la cantidad de barras en el histograma y si se agrega una estimación de densidad de kernel, respectivamente. Se establecen los títulos y etiquetas adecuados antes de mostrar la figura con `plt.show()`. Asegúrate de tener instaladas las bibliotecas Seaborn y Matplotlib antes de ejecutar este código. Puedes instalarlas usando los comandos `pip install seaborn matplotlib`.

La grafica obtenida al ejecutar el código es:



### 3.1.8 Gráfica de la correlación entre las columnas Height y Weight

Solicita a CHATGPT que genere el código que grafique la correlación entre las columnas Height y Weight usando Seaborn

Solicitud:



Genera el código Python que grafique la correlación entre las columnas Height y Weight, de la información anterior, usando Seaborn

Resultado:



Por supuesto, aquí tienes el código en Python que utiliza la biblioteca Seaborn para graficar la correlación entre las columnas "Height" (Altura) y "Weight" (Peso) del DataFrame:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"
data = pd.read_csv(url)

sns.set(style="whitegrid") # Configura el estilo de Seaborn

plt.figure(figsize=(8, 6)) # Tamaño de la figura

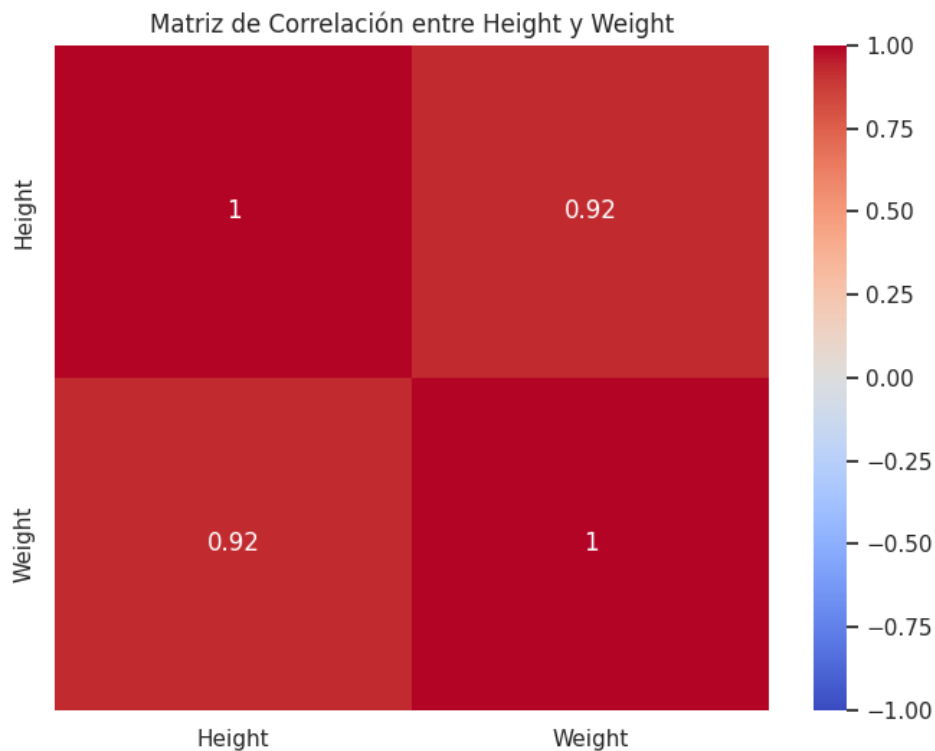
# Crea la matriz de correlación utilizando Seaborn
correlation_matrix = data[['Height', 'Weight']].corr()
sns.heatmap(correlation_matrix, annot=True, cmap="coolwarm", vmin=-1, vmax=1)

plt.title('Matriz de Correlación entre Height y Weight')
plt.show()
```

Este código crea una matriz de correlación entre las columnas "Height" y "Weight" utilizando la función `.corr()` de Pandas y luego la visualiza utilizando un mapa de calor (heatmap) proporcionado por Seaborn. El mapa de calor muestra los valores de correlación entre las dos columnas. Los parámetros `annot`, `cmap`, `vmin` y `vmax` controlan la visualización de los valores y el color en el mapa de calor. Se establecen títulos y etiquetas adecuados antes de mostrar la figura con `plt.show()`. Asegúrate

de tener instaladas las bibliotecas Seaborn y Matplotlib antes de ejecutar este código. Puedes instalarlas usando los comandos `pip install seaborn matplotlib`.

La grafica obtenida al ejecutar el código es:



### 3.1.9 Correlación entre las columnas `Height` y `Weight` usando `corr` de Numpy

Solicita a CHATGPT que genere el código que obtenga la correlación entre las columnas `Height` y `Weight` usando `corr` de Numpy

Solicitud:

CI

Genera el código Python que obtenga la correlación entre las columnas `Height` y `Weight`, de la información anterior, usando `corr` de Numpy

## Resultado:



Por supuesto, aquí tienes el código en Python que utiliza la biblioteca NumPy para calcular la correlación entre las columnas "Height" (Altura) y "Weight" (Peso) del DataFrame:

```
import pandas as pd
import numpy as np

url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"
data = pd.read_csv(url)

correlation = np.corrcoef(data['Height'], data['Weight'])[0, 1]
print("Correlation between Height and Weight:", correlation)
```

## Resultado de ejecutar el código obtenido:

```
Correlation between Height and Weight: 0.9247562987409146
```

### 3.1.10 Valores promedio de las columnas weight y Height por genero

Solicita a CHATGPT que genere el código que reporte el valor promedio de la columna Height donde Gender es Male y donde Gender es Female y repita lo mismo para la columna Weight

## Solicitud:



Genera el código Python que reporte el valor promedio de la columna Height donde Gender es Male y donde Gender es Female y repite lo mismo para la columna Weight, de la información anterior

## Resultado:



Por supuesto, aquí tienes el código en Python que calcula el valor promedio de las columnas "Height" (Altura) y "Weight" (Peso) para los géneros "Male" y "Female" en el DataFrame:

```
import pandas as pd

url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"
```



```
data = pd.read_csv(url)

male_height_mean = data[data['Gender'] == 'Male']['Height'].mean()
female_height_mean = data[data['Gender'] == 'Female']['Height'].mean()

male_weight_mean = data[data['Gender'] == 'Male']['Weight'].mean()
female_weight_mean = data[data['Gender'] == 'Female']['Weight'].mean()

print("Average Height for Male:", male_height_mean)
print("Average Height for Female:", female_height_mean)
print("Average Weight for Male:", male_weight_mean)
print("Average Weight for Female:", female_weight_mean)
```

Este código filtra el DataFrame para las entradas donde el género es "Male" y luego calcula el valor promedio de las columnas "Height" y "Weight". Luego, hace lo mismo para el género "Female". Los valores promedio se imprimen en la salida. Asegúrate de tener instalada la biblioteca Pandas antes de ejecutar este código, la puedes instalar usando el comando `pip install pandas`.

Resultado de ejecutar el código obtenido:

```
Average Height for Male: 69.02634590621741
Average Height for Female: 63.70877360342507
Average Weight for Male: 187.0206206581932
Average Weight for Female: 135.86009300746835
```

## 3.2 EXTRACCIÓN DE INFORMACIÓN

La extracción de texto de una imagen, también conocida como OCR (Optical Character Recognition, por sus siglas en inglés), es una tecnología que consiste en convertir el contenido textual presente en una imagen o un documento escaneado en texto digital editable. Esto implica identificar y reconocer los caracteres individuales, palabras y frases en la imagen para luego convertirlos en texto legible que puede ser copiado, editado y buscado en sistemas informáticos.

La extracción de texto de una imagen tiene varias aplicaciones y es de gran importancia en diversos campos. A continuación, veremos algunos ejemplos de aplicación.

### 3.2.1 Extraer el texto de una imagen PNG

Solicita a CHATGPT que genere el código que extraiga el texto de una imagen PNG usando OCR en Python desde la URL [https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/UITD\\_PAPER.png](https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/UITD_PAPER.png)

**Solicitud:**



Estoy usando Google Colab, Genera el código en Python usando OCR que extraiga el texto de la imagen PNG desde una URL de internet determinada

**Resultado:**



Por supuesto, puedo ayudarte con eso. Para extraer texto de una imagen PNG desde una URL de internet usando OCR en Google Colab, puedes utilizar la biblioteca `pytesseract` junto con `Pillow` para cargar y procesar la imagen. Asegúrate de que tienes `pytesseract` instalado y de que Tesseract OCR está disponible en tu entorno Colab.

Aquí tienes un ejemplo de cómo podrías hacerlo:

```
# Instala las bibliotecas necesarias
!pip install pytesseract
!pip install Pillow
```

```

# Importa las bibliotecas
from PIL import Image
import requests
import pytesseract

# URL de la imagen PNG
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/UITD_PAPER.png"

# Descarga la imagen desde la URL
response = requests.get(url)
image = Image.open(BytesIO(response.content))

# Utiliza pytesseract para extraer el texto de la imagen
text = pytesseract.image_to_string(image)

# Imprime el texto extraído
print(text)

```

Asegúrate de reemplazar "URL\_DE\_LA\_IMAGEN" con la URL real de la imagen PNG que deseas procesar. Ten en cuenta que la precisión del OCR puede variar según la calidad y la legibilidad de la imagen, así como el contenido del texto en sí. Recuerda que para ejecutar este código en Google Colab, necesitarás una conexión a internet y las bibliotecas mencionadas deben estar disponibles en tu entorno Colab.

Resultado de ejecutar el código obtenido:

```

1. "Saving Time with the User Interface Transition Diagrams (UITD) Editor"
2. "The UITD Editor: A Tool to Simplify User-System Interaction Modeling"
3. "Streamlining User Interface Design with the UITD Editor: A Comparative Study"

4. "UITD Editor: Simplifying User-System Interaction Modeling for Developers"

Can the User Interface Transition Diagrams editor (UITD editor) save users' time?

Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.
Department of Applied Mathematics and Systems
Universidad Autonoma Metropolitana
México City, México
jcervantes@.cua.uam.mx, dragonnomadal23@gmail.com, mgomez@.cua.uam.mx

```

**Abstract**— The User Interface Transition Diagram (UITD) is a formal modeling notation that simplifies the specification and design of user-system interactions. It is a valuable communication tool for technical and non-technical stakeholders during the requirements elicitation phase, as it provides a simple and complete notation that is easy to understand. In this paper, we investigate the efficiency of creating UITDs using draw.io, a widely used diagramming software, compared to a dedicated UITD editor. We conducted a study to compare the time required to complete the task, as well as the subjective ease of use and satisfaction of participants. Our results show that the UITD editor is more efficient and preferred by participants, highlighting the importance of using specialized tools for creating formal models such as UITDs. The findings of this study have implications for software developers, designers, and other stakeholders involved in the specification and design of user-system interactions.

Editor Efficiency; Modelling notation; Requirements elicitation

## I. INTRODUCTION

The User Interface Transition Diagram (UITD) is a formal modeling notation that simplifies the specification and design of user-system interactions. It can be used to model the flow of user interfaces that the system will have, making it a valuable communication tool for technical and non-technical stakeholders during the requirements elicitation phase. Additionally, it is technically accurate, allowing it to be used reliably to start the development of the system [1].

Simplicity is key when working with modeling notations, particularly for non-experts [3]. The UITD provides a notation that is both complete and simple enough for non-technical stakeholders to understand [2]. It is a formal modeling notation that is easy to learn, facilitating communication between stakeholders with and without software development technical skills. Compared to other formal modeling tools, the UITD has several advantages, including its simplicity, completeness, and ability to model user-system interactions in a clear and concise way. Empirical evidence about the UITD's understandability by non-technical stakeholders is also provided in [2]. By using the UITD, designers and developers can create models that are both technically accurate and easy to understand, helping to ensure that all stakeholders are aligned and satisfied with the specification of the system's requirements.

In recent years, various tools have been developed to support the creation of generalized diagrams. One such tool is draw.io, which is a well-known open-source diagramming software. Draw.io is widely used for creating diagrams, flowcharts, and other types of graphical representations. While draw.io is a versatile tool, it may not be specifically designed to handle the requirements of creating UITDs. This raises the question of how easy it is to create UITDs using draw.io, and whether it is more efficient to use a dedicated tool such as the UITD editor [4].

To investigate this question, we conducted a study in which participants were asked to create UITDs using both draw.io and the UITD editor. Our study aimed to compare the efficiency of the two tools in

En este ejercicio debimos especificar que se está usando Google Colab para que el código funcionara de forma directa. A sí mismo también tuve que indicar a la solicitud de ChatGPT que se trata de extraer el texto de una imagen PNG que se encuentra en una URL determinada, que ya si agregaba directamente la URL específica, la respuesta era que no podía tener acceso a internet. Al final de la respuesta el chat indica que no hay que olvidar que se debe colocar en el código la URL de la imagen para que funcione correctamente. De esta forma el resultado fue satisfactorio.

Aquí tenemos la imagen original:

1. "Saving Time with the User Interface Transition Diagrams (UITD) Editor"
2. "The UITD Editor: A Tool to Simplify User-System Interaction Modeling"
3. "Streamlining User Interface Design with the UITD Editor: A Comparative Study"
4. "UITD Editor: Simplifying User-System Interaction Modeling for Developers"

### Can the User Interface Transition Diagrams editor (UITD editor) save users' time?

Cervantes-Ojeda J., Badillo-Salas A., Gómez-Fuentes M.C.

Department of Applied Mathematics and Systems

Universidad Autónoma Metropolitana

México City, México

jcervantes@cua.uam.mx, dragonnomada123@gmail.com, mgomez@cua.uam.mx

**Abstract**— The User Interface Transition Diagram (UITD) is a formal modeling notation that simplifies the specification and design of user-system interactions. It is a valuable communication tool for technical and non-technical stakeholders during the requirements elicitation phase, as it provides a simple and complete notation that is easy to understand. In this paper, we investigate the efficiency of creating UITDs using draw.io, a widely used diagramming software, compared to a dedicated UITD editor. We conducted a study to compare the time required to complete the task, as well as the subjective ease of use and satisfaction of participants. Our results show that the UITD editor is more efficient and preferred by participants, highlighting the importance of using specialized tools for creating formal models such as UITDs. The findings of this study have implications for software developers, designers, and other stakeholders involved in the specification and design of user-system interactions.

*Editor Efficiency; Modelling notation; Requirements elicitation*

#### I. INTRODUCTION

The User Interface Transition Diagram (UITD) is a formal modeling notation that simplifies the specification and design of user-system interactions. It can be used to model the flow of user interfaces that the system will have, making it a valuable communication tool for technical and non-technical stakeholders during the requirements elicitation phase. Additionally, it is technically accurate, allowing it to be used reliably to start the development of the system [1].

Simplicity is key when working with modeling notations, particularly for non-experts [3]. The UITD provides a notation that is both complete and simple enough for non-technical stakeholders to understand [2]. It is a formal modeling notation that is easy to learn, facilitating communication between stakeholders with and without software development technical skills. Compared to other formal modeling tools, the UITD has several advantages, including its simplicity, completeness, and ability to model user-system interactions in a clear and concise way. Empirical evidence about the UITD's understandability by non-technical stakeholders is also provided in [2]. By using the UITD, designers and developers can create models that are both technically accurate and easy to understand, helping to ensure that all stakeholders are aligned and satisfied with the specification of the system's requirements.

In recent years, various tools have been developed to support the creation of generalized diagrams. One such tool is draw.io, which is a well-known open-source diagramming software. Draw.io is widely used for creating diagrams, flowcharts, and other types of graphical representations. While draw.io is a versatile tool, it may not be specifically designed to handle the requirements of creating UITDs. This raises the question of how easy it is to create UITDs using draw.io, and whether it is more efficient to use a dedicated tool such as the UITD editor [4].

To investigate this question, we conducted a study in which participants were asked to create UITDs using both draw.io and the UITD editor. Our study aimed to compare the efficiency of the two tools in

## 3.2.2 Traducción al español

Solicita a CHATGPT que traduzca el texto a español (pega el texto extraído en <https://chat.openai.com/>)

## Solicitud:

CI

traduce el siguiente texto a español:

1. "Saving Time with the User Interface Transition Diagrams (UITD) Editor"
2. "The UITD Editor: A Tool to Simplify User-System Interaction Modeling"
3. "Streamlining User Interface Design with the UITD Editor: A Comparative Study"
4. "UITD Editor: Simplifying User-System Interaction Modeling for Developers"

Can the User Interface Transition Diagrams editor (UITD editor) save users' time?

Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.

Department of Applied Mathematics and Systems

Universidad Autonoma Metropolitana

México City, México

[jcervantes@cua.uam.mx](mailto:jcervantes@cua.uam.mx), [dragonnomadal23@gmail.com](mailto:dragonnomadal23@gmail.com), [mgomez@cua.uam.mx](mailto:mgomez@cua.uam.mx)

**Abstract**— The User Interface Transition Diagram (UITD) is a formal modeling notation that simplifies the specification and design of user-system interactions. It is a valuable communication tool for technical and non-technical stakeholders during the requirements elicitation phase, as it provides a simple and complete notation that is easy to understand. In this paper, we investigate the efficiency of creating UITDs using draw.io, a widely used diagramming software, compared to a dedicated UITD editor. We conducted a study to compare the time required to complete the task, as well as the subjective ease of use and satisfaction of participants. Our results show that the UITD editor is more efficient and preferred by participants, highlighting the importance of using specialized tools for creating formal models such as UITDs. The findings of this study have implications for software developers, designers, and other stakeholders involved in the specification and design of user-system interactions.

Editor Efficiency; Modelling notation; Requirements elicitation

## I. INTRODUCTION

The User Interface Transition Diagram (UITD) is a formal modeling notation that simplifies the specification and design of user-system interactions. It can be used to model the flow of user interfaces that the system will have, making it a valuable communication tool for technical and non-technical stakeholders during the requirements elicitation

phase. Additionally, it is technically accurate, allowing it to be used reliably to start the development of the system [1].

Simplicity is key when working with modeling notations, particularly for non-experts [3]. The UITD provides a notation that is both complete and simple enough for non-technical stakeholders to understand [2]. It is a formal modeling notation that is easy to learn, facilitating communication between stakeholders with and without software development technical skills. Compared to other formal modeling tools, the UITD has several advantages, including its simplicity, completeness, and ability to model user-system interactions in a clear and concise way. Empirical evidence about the UITD's understandability by non-technical stakeholders is also provided in [2]. By using the UITD, designers and developers can create models that are both technically accurate and easy to understand, helping to ensure that all stakeholders are aligned and satisfied with the specification of the system's requirements.

In recent years, various tools have been developed to support the creation of generalized diagrams. One such tool is draw.io, which is a well-known open-source diagramming software. Draw.io is widely used for creating diagrams, flowcharts, and other types of graphical representations. While draw.io is a versatile tool, it may not be specifically designed to handle the requirements of creating UITDs. This raises the question of how easy it is to create UITDs using draw.io, and whether it is more efficient to use a dedicated tool such as the UITD editor [4].

To investigate this question, we conducted a study in which participants were asked to create UITDs using both draw.io and the UITD editor. Our study aimed to compare the efficiency of the two tools in

## Resultado:



1. "Ahorro de tiempo con el Editor de Diagramas de Transición de Interfaz de Usuario (UITD)"
2. "El Editor UITD: Una herramienta para simplificar la modelización de interacciones entre usuario y sistema"
3. "Optimización del diseño de la interfaz de usuario con el Editor UITD: Un estudio comparativo"



#### 4. "Editor UITD: Simplificando la modelización de interacciones entre usuario y sistema para desarrolladores"

¿Puede el editor de Diagramas de Transición de Interfaz de Usuario (editor UITD) ahorrar tiempo a los usuarios?

Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C. Departamento de Matemáticas Aplicadas y Sistemas Universidad Autónoma Metropolitana Ciudad de México, México [jcervantes@cua.uam.mx](mailto:jcervantes@cua.uam.mx), [dragonnomadal23@gmail.com](mailto:dragonnomadal23@gmail.com), [mgomez@cua.uam.mx](mailto:mgomez@cua.uam.mx)

Resumen— El Diagrama de Transición de Interfaz de Usuario (UITD) es una notación de modelado formal que simplifica la especificación y el diseño de interacciones entre el usuario y el sistema. Es una herramienta de comunicación valiosa para partes interesadas técnicas y no técnicas durante la fase de obtención de requisitos, ya que proporciona una notación simple y completa que es fácil de entender. En este artículo, investigamos la eficiencia de crear UITDs utilizando draw.io, un software de diagramación ampliamente utilizado, en comparación con un editor UITD dedicado. Realizamos un estudio para comparar el tiempo necesario para completar la tarea, así como la facilidad de uso subjetiva y la satisfacción de los participantes. Nuestros resultados muestran que el editor UITD es más eficiente y preferido por los participantes, resaltando la importancia de usar herramientas especializadas para crear modelos formales como los UITDs. Los hallazgos de este estudio tienen implicaciones para desarrolladores de software, diseñadores y otras partes interesadas involucradas en la especificación y el diseño de interacciones entre el usuario y el sistema.

Eficiencia del editor; Notación de modelado; Obtención de requisitos

#### I. INTRODUCCIÓN

El Diagrama de Transición de Interfaz de Usuario (UITD) es una notación de modelado formal que simplifica la especificación y el diseño de interacciones entre el usuario y el sistema. Puede usarse para modelar el flujo de las interfaces de usuario que tendrá el sistema, lo que lo convierte en una herramienta de comunicación valiosa para partes interesadas técnicas y no técnicas durante la fase de obtención de requisitos. Además, es técnicamente preciso, lo que permite utilizarlo de manera confiable para iniciar el desarrollo del sistema [1].

La simplicidad es clave al trabajar con notaciones de modelado, especialmente para no expertos [3]. El UITD proporciona una notación que es completa y lo suficientemente simple como para que las partes interesadas no técnicas la comprendan [2]. Es una notación de modelado formal que es fácil de aprender, lo que facilita la comunicación entre partes interesadas con y sin habilidades técnicas en el desarrollo de software. En comparación con otras herramientas de modelado formal, el UITD tiene varias ventajas, incluyendo su simplicidad, completitud y capacidad para modelar interacciones entre el usuario y el sistema de manera clara y concisa. También se proporciona evidencia empírica sobre la comprensibilidad del

UITD por parte de las partes interesadas no técnicas en [2]. Al usar el UITD, los diseñadores y desarrolladores pueden crear modelos que son técnicamente precisos y fáciles de entender, lo que ayuda a garantizar que todas las partes interesadas estén alineadas y satisfechas con la especificación de los requisitos del sistema.

En los últimos años, se han desarrollado diversas herramientas para admitir la creación de diagramas generalizados. Una de estas herramientas es draw.io, que es un conocido software de diagramación de código abierto. Draw.io se usa ampliamente para crear diagramas, flujogramas y otros tipos de representaciones gráficas. Si bien draw.io es una herramienta versátil, es posible que no esté diseñada específicamente para manejar los requisitos de creación de UITDs. Esto plantea la pregunta de cuán fácil es crear UITDs usando draw.io y si es más eficiente usar una herramienta dedicada como el editor UITD [4].

Para investigar esta pregunta, realizamos un estudio en el que se pidió a los participantes que crearan UITDs utilizando tanto draw.io como el editor UITD. Nuestro estudio tenía como objetivo comparar la eficiencia de las dos herramientas en... (El texto se corta aquí debido a la limitación de caracteres en la respuesta. Si necesitas más traducción, no dudes en pedirla).

### 3.2.3 Autores del Artículo

Solicita a CHATGPT que te diga quiénes son los autores del artículo

**Solicitud:**



De la traducción anterior dime quienes son los autores

**Resultado:**



Los autores mencionados en el texto son:

- Cervantes-Ojeda J.
- Badillo-Salas A.
- Gomez-Fuentes M.C.

Estos autores pertenecen al Departamento de Matemáticas Aplicadas y Sistemas de la Universidad Autónoma Metropolitana en Ciudad de México, México. Sus correos electrónicos son: [jcervantes@cua.uam.mx](mailto:jcervantes@cua.uam.mx), [dragonnomadal23@gmail.com](mailto:dragonnomadal23@gmail.com), [mgomez@cua.uam.mx](mailto:mgomez@cua.uam.mx).

### 3.2.4 Extracción de correos electrónicos

Solicita a CHATGPT que te diga cuáles son los correos relacionados al artículo

**Solicitud:**



De la traducción anterior dime cuales son los correos electrónicos relacionados

**Resultado:**



Los correos electrónicos relacionados con los autores son:

- [jcervantes@cua.uam.mx](mailto:jcervantes@cua.uam.mx)
- [dragonnomadal23@gmail.com](mailto:dragonnomadal23@gmail.com)
- [mgomez@cua.uam.mx](mailto:mgomez@cua.uam.mx)

### 3.2.5 Extracción de País, Ciudad, Departamento y Universidad

Solicita a CHATGPT que te diga quién es la universidad, el departamento, la ciudad y el país del artículo

**Solicitud:**



De la traducción anterior dime cual es la universidad, el departamento, la ciudad y el país del artículo

**Resultado:**



La universidad, el departamento, la ciudad y el país del artículo son los siguientes:

- Universidad: Universidad Autónoma Metropolitana
- Departamento: Departamento de Matemáticas Aplicadas y Sistemas
- Ciudad: Ciudad de México
- País: México

### 3.2.6 Explicación del ABSTRACT

Solicita a CHATGPT que te explique el ABSTRACT

Solicitud:



Explica la sección “resumen” de la traducción anterior

Resultado:



Claro, con gusto. La sección de "Resumen" en la traducción anterior del artículo proporciona una descripción concisa de los aspectos más relevantes del estudio y su contexto. Aquí está una explicación más detallada de cada parte del resumen:

- **Contexto y Problema Abordado:** El resumen comienza presentando el contexto en el que se desarrolla el estudio. Destaca que el "Diagrama de Transición de Interfaz de Usuario" (UITD) es una notación formal de modelado. Esta notación es utilizada para simplificar la especificación y diseño de las interacciones entre los usuarios y los sistemas. Al mencionar "simplificación", se sugiere que el proceso puede ser complejo y que el UITD ayuda a hacerlo más comprensible.
- **Importancia del UITD:** El resumen subraya que el UITD es especialmente valioso durante la fase de obtención de requisitos. En esta etapa inicial del desarrollo de software, tanto las partes interesadas técnicas como las no técnicas deben entender cómo interactuarán los usuarios con el sistema. El UITD se presenta como una herramienta de comunicación esencial para lograr esta comprensión y garantizar que todos los involucrados estén en la misma página.
- **Objetivo del Estudio:** El resumen declara el objetivo principal del estudio: comparar la eficiencia de dos herramientas diferentes para crear UITDs. La primera herramienta es "draw.io", una herramienta de diagramación ampliamente conocida pero no específicamente diseñada para UITDs. La segunda herramienta es un editor dedicado de UITD. La meta es evaluar cuál de estas opciones es más efectiva en términos de tiempo de creación, facilidad de uso y satisfacción de los usuarios.
- **Resultados Relevantes:** El resumen anticipa los resultados obtenidos en el estudio al afirmar que el editor de UITD resultó más eficiente y preferido por los participantes en comparación con "draw.io". Esto resalta la importancia de contar con herramientas diseñadas específicamente para tareas particulares, como la creación de UITDs. Se subraya que esta conclusión tiene implicaciones para diversos roles en el desarrollo de software, incluyendo a desarrolladores y diseñadores, así como para aquellos que participan en la especificación y diseño de interacciones usuario-sistema.

En resumen, la sección de "Resumen" del artículo brinda una visión general del contexto, el objetivo y los resultados del estudio, destacando la importancia del UITD como herramienta de modelado y comunicación, así como la eficiencia del editor de UITD en comparación con herramientas más generales.

### 3.2.7 Explicación de la introducción

Solicita a CHATGPT que te explique la introducción

**Solicitud:**



Explícame la introducción de la traducción anterior

**Resultado:**



Por supuesto, aquí tienes una explicación de la introducción de la traducción anterior: La introducción del artículo establece el contexto y presenta la importancia del "Diagrama de Transición de Interfaz de Usuario" (UITD) como una notación formal de modelado. El UITD es destacado como una herramienta valiosa en la especificación y diseño de interacciones entre usuarios y sistemas. Se señala que esta notación es especialmente útil en la fase inicial de obtención de requisitos, donde las partes técnicas y no técnicas deben comprender cómo se relacionarán los usuarios con el sistema.

La introducción destaca la relevancia de la simplicidad en las notaciones de modelado, especialmente para personas que no son expertas en tecnología. Se menciona que el UITD aborda esta necesidad al proporcionar una notación que es completa pero también fácil de entender para las partes interesadas no técnicas. Además, se destaca que el UITD es técnicamente preciso, lo que lo convierte en una herramienta confiable para iniciar el desarrollo del sistema.

El objetivo del artículo se establece al plantear una pregunta: dado que existen herramientas generales como "draw.io" para crear diagramas y herramientas específicas como el editor UITD, ¿cuál es la eficiencia y la eficacia de estas herramientas al crear UITDs? La pregunta sobre la eficacia y la eficiencia se enmarca en la comparación entre la herramienta general y la herramienta específica.

En resumen, la introducción del artículo presenta el contexto del UITD como una notación de modelado valiosa en el diseño de interacciones usuario-sistema. También resalta la necesidad de simplicidad en las notaciones para facilitar la comprensión. Finalmente, plantea la pregunta central del estudio: cómo se comparan las herramientas generales y específicas en términos de eficacia y eficiencia en la creación de UITDs

### 3.2.8 Extracción de las palabras principales

Solicita a CHATGPT que extraiga las palabras principales

**Solicitud:**



De la anterior traducción de un artículo dame las palabras principales

**Resultado:**



Aquí están las palabras principales de la traducción del artículo:

- Ahorro de Tiempo
- Editor de Diagramas de Transición de Interfaz de Usuario (UITD)
- Herramienta
- Simplificar
- Modelización de Interacción Usuario-Sistema
- Optimización
- Diseño de Interfaz de Usuario
- Estudio Comparativo
- Editor
- Desarrolladores
- Diagrama de Transición de Interfaz de Usuario (UITD)
- Modelado
- Notación
- Comunicación
- Técnico
- No Técnico
- Obtención de Requisitos
- Universidad Autónoma Metropolitana
- Ciudad de México
- Resumen
- Eficacia
- Participantes
- Eficiencia
- Simplicidad
- Completo
- Comunicación
- Partes Interesadas
- Tarea
- Satisfacción
- Resultados

- Importancia
- Herramientas Especializadas
- Desarrollo de Software
- Diseñadores
- Implicaciones
- Especificación
- Draw.io
- Software de Diagramación
- Flujogramas
- Representaciones Gráficas
- Versátil
- Estudio
- Comparación

### 3.2.9 Reescribir un texto en formato Markdown

Solicita a CHATGPT que genere el código para que reescriba el artículo en formato de Markdown

**Solicitud:**

CI

Genera el código Python para reescribir la anterior traducción de un artículo en formato de Markdown

**Resultado:**



Aquí tienes el artículo reescrito en formato de Markdown usando código Python:

```
article_info = {
    "title": "Ahorro de Tiempo con el Editor de Diagramas de Transición de Interfaz de
Usuario (UITD)",
    "authors": "Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.",
    "affiliation": "Departamento de Matemáticas Aplicadas y Sistemas, Universidad
Autónoma Metropolitana, Ciudad de México, México",
    "contact": "Contacto: jcervantes@cua.uam.mx, dragonnomadal23@gmail.com,
mgomez@cua.uam.mx",
}

abstract = """
```

## ## Resumen

El **Diagrama de Transición de Interfaz de Usuario (UITD)** es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. En este artículo, investigamos la eficiencia de crear UITDs utilizando *draw.io*, un software de diagramación ampliamente utilizado, en comparación con un editor UITD dedicado. Realizamos un estudio para comparar el tiempo requerido para completar la tarea, así como la facilidad de uso subjetiva y la satisfacción de los participantes. Nuestros resultados muestran que el editor UITD es más eficiente y preferido por los participantes, resaltando la importancia de utilizar herramientas especializadas para crear modelos formales como los UITDs. Los hallazgos de este estudio tienen implicaciones para desarrolladores de software, diseñadores y otras partes interesadas involucradas en la especificación y diseño de interacciones usuario-sistema.

**Palabras Clave:** Editor Eficiencia; Notación de Modelado; Obtención de Requisitos

introduction = ""

## ## Introducción

El **UITD** es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. Se puede utilizar para modelar el flujo de interfaces de usuario que tendrá el sistema, convirtiéndose en una herramienta valiosa durante la fase de obtención de requisitos. Además, es técnicamente preciso, lo que permite su uso confiable para iniciar el desarrollo del sistema.

La simplicidad es clave al trabajar con notaciones de modelado, especialmente para no expertos. El UITD proporciona una notación completa y comprensible para las partes interesadas no técnicas, facilitando la comunicación entre diferentes grupos. En comparación con otras herramientas de modelado formal, el UITD tiene ventajas como su simplicidad, completitud y capacidad para representar de manera clara y concisa las interacciones usuario-sistema. Además, hay evidencia empírica sobre su comprensibilidad por parte de las partes no técnicas. Al usar el UITD, los diseñadores y desarrolladores pueden crear modelos técnicamente precisos y fáciles de entender, asegurando que todas las partes estén alineadas y satisfechas con la especificación de requisitos del sistema.

En los últimos años, se han desarrollado diversas herramientas para respaldar la creación de diagramas en general. Un ejemplo es *draw.io*, un software de diagramación de código abierto ampliamente utilizado. Aunque *draw.io* es versátil, puede que no esté específicamente diseñado para manejar los requisitos de creación de UITDs. Esto plantea la pregunta de cuán fácil es crear UITDs con *draw.io* y si es más eficiente usar una herramienta dedicada como el editor UITD.



```
Para investigar esto, realizamos un estudio en el que los participantes crearon UITDs usando tanto *draw.io* como el editor UITD. Nuestro estudio buscaba comparar la eficiencia de ambas herramientas...
```

```
"""
```

```
markdown_article = f"""
```

```
# {article_info['title']}
```

```
*Autores*: {article_info['authors']}
```

```
*Afiliación*: {article_info['affiliation']}
```

```
*Contacto*: {article_info['contact']}
```

```
{abstract}
```

```
{introduction}
```

```
"""
```

```
print(markdown_article)
```

Este código generará el artículo reescrito en formato de Markdown. Puedes ejecutar el código y ajustar el contenido según sea necesario.

Resultado de ejecutar el código obtenido:

```
# Ahorro de Tiempo con el Editor de Diagramas de Transición de Interfaz de Usuario (UITD)
```

```
*Autores*: Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.
```

```
*Afiliación*: Departamento de Matemáticas Aplicadas y Sistemas,  
Universidad Autónoma Metropolitana, Ciudad de México, México
```

```
*Contacto*: Contacto: jcervantes@cua.uam.mx,  
dragonnomadal23@gmail.com, mgomez@cua.uam.mx
```

```
## Resumen
```

El **Diagrama de Transición de Interfaz de Usuario (UITD)** es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. En este artículo, investigamos la eficiencia de crear UITDs utilizando *\*draw.io\**, un software de diagramación ampliamente utilizado, en comparación con un editor UITD dedicado. Realizamos un estudio para comparar el tiempo requerido para completar la tarea, así como la facilidad de uso subjetiva y la satisfacción de los participantes. Nuestros resultados muestran que el editor UITD es más eficiente y preferido por los participantes, resaltando la importancia de utilizar herramientas especializadas para crear modelos formales como los UITDs. Los hallazgos de este estudio tienen implicaciones para desarrolladores de

software, diseñadores y otras partes interesadas involucradas en la especificación y diseño de interacciones usuario-sistema.

**\*\*Palabras Clave\*\*:** Editor Eficiencia; Notación de Modelado; Obtención de Requisitos

## ## Introducción

El **\*\*UITD\*\*** es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. Se puede utilizar para modelar el flujo de interfaces de usuario que tendrá el sistema, convirtiéndose en una herramienta valiosa durante la fase de obtención de requisitos. Además, es técnicamente preciso, lo que permite su uso confiable para iniciar el desarrollo del sistema.

La simplicidad es clave al trabajar con notaciones de modelado, especialmente para no expertos. El UITD proporciona una notación completa y comprensible para las partes interesadas no técnicas, facilitando la comunicación entre diferentes grupos. En comparación con otras herramientas de modelado formal, el UITD tiene ventajas como su simplicidad, completitud y capacidad para representar de manera clara y concisa las interacciones usuario-sistema. Además, hay evidencia empírica sobre su comprensibilidad por parte de las partes no técnicas. Al usar el UITD, los diseñadores y desarrolladores pueden crear modelos técnicamente precisos y fáciles de entender, asegurando que todas las partes estén alineadas y satisfechas con la especificación de requisitos del sistema.

En los últimos años, se han desarrollado diversas herramientas para respaldar la creación de diagramas en general. Un ejemplo es *\*draw.io\**, un software de diagramación de código abierto ampliamente utilizado. Aunque *\*draw.io\** es versátil, puede que no esté específicamente diseñado para manejar los requisitos de creación de UITDs. Esto plantea la pregunta de cuán fácil es crear UITDs con *\*draw.io\** y si es más eficiente usar una herramienta dedicada como el editor UITD.

Para investigar esto, realizamos un estudio en el que los participantes crearon UITDs usando tanto *\*draw.io\** como el editor UITD. Nuestro estudio buscaba comparar la eficiencia de ambas herramientas...

### 3.2.10 Reescribir artículo en forma Latex

Solicita a CHATGPT que genere el código para que reescriba el artículo en forma Latex

**Solicitud:**

CI

Genera el código Python para reescribir la anterior traducción de un artículo en forma LATEX

## Resultado:



Aquí tienes el artículo reescrito en formato de LaTeX usando código Python:

```
article_info = {  
    "title": "Ahorro de Tiempo con el Editor de Diagramas de Transición de  
Interfaz de Usuario (UITD)",  
    "authors": "Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.",  
    "affiliation": "Departamento de Matemáticas Aplicadas y Sistemas,  
Universidad Autónoma Metropolitana, Ciudad de México, México",  
    "contact": "Contacto: jcervantes@cua.uam.mx, dragonnomadal23@gmail.com,  
mgomez@cua.uam.mx",  
}
```

```
abstract = r"""  
\section{Resumen}
```

El \textbf{Diagrama de Transición de Interfaz de Usuario (UITD)} es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. En este artículo, investigamos la eficiencia de crear UITDs utilizando \textit{draw.io}, un software de diagramación ampliamente utilizado, en comparación con un editor UITD dedicado. Realizamos un estudio para comparar el tiempo requerido para completar la tarea, así como la facilidad de uso subjetiva y la satisfacción de los participantes. Nuestros resultados muestran que el editor UITD es más eficiente y preferido por los participantes, resaltando la importancia de utilizar herramientas especializadas para crear modelos formales como los UITDs. Los hallazgos de este estudio tienen implicaciones para desarrolladores de software, diseñadores y otras partes interesadas involucradas en la especificación y diseño de interacciones usuario-sistema.

```
\textbf{Palabras Clave:} Editor Eficiencia; Notación de Modelado; Obtención  
de Requisitos  
"""
```

```
introduction = r"""  
\section{Introducción}
```

El \textbf{UITD} es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. Se puede utilizar para modelar el flujo de interfaces de usuario que tendrá el sistema, convirtiéndose en una herramienta valiosa durante la fase de obtención de requisitos. Además, es técnicamente preciso, lo que permite su uso confiable para iniciar el desarrollo del sistema.

La simplicidad es clave al trabajar con notaciones de modelado, especialmente para no expertos. El UITD proporciona una notación completa y comprensible para las partes interesadas no técnicas, facilitando la comunicación entre diferentes grupos. En comparación con otras herramientas de modelado formal, el UITD tiene ventajas como su simplicidad, completitud y capacidad para representar de manera clara y concisa las interacciones usuario-sistema. Además, hay evidencia empírica sobre su comprensibilidad por parte de las partes no técnicas. Al usar el UITD, los diseñadores y desarrolladores pueden crear modelos técnicamente precisos y fáciles de entender, asegurando que todas las partes estén alineadas y satisfechas con la especificación de requisitos del sistema.

En los últimos años, se han desarrollado diversas herramientas para respaldar la creación de diagramas en general. Un ejemplo es `\textit{draw.io}`, un software de diagramación de código abierto ampliamente utilizado. Aunque `\textit{draw.io}` es versátil, puede que no esté específicamente diseñado para manejar los requisitos de creación de UITDs. Esto plantea la pregunta de cuán fácil es crear UITDs con `\textit{draw.io}` y si es más eficiente usar una herramienta dedicada como el editor UITD.

Para investigar esto, realizamos un estudio en el que los participantes crearon UITDs usando tanto `\textit{draw.io}` como el editor UITD. Nuestro estudio buscaba comparar la eficiencia de ambas herramientas...

"""

```
latex_article = fr"""
\documentclass{{article}}
\title{{{{article_info['title']}}}
\author{{{{article_info['authors']}}}
\date{{{{}}
\begin{{document}}
\maketitle
\section*{{A filiación}}
{{article_info['affiliation']}}
\section*{{Contacto}}
{{article_info['contact']}}

{abstract}

{introduction}

\end{{document}}
"""
```

```
print(latex_article)
```

Este código generará el artículo reescrito en formato de LaTeX. Puedes ejecutar el código y ajustar el contenido según sea necesario. Ten en cuenta que el código LaTeX puede necesitar un compilador LaTeX para generar un documento PDF a partir del código resultante.

Resultado de ejecutar el código obtenido:

```
\documentclass{article}
\title{Ahorro de Tiempo con el Editor de Diagramas de Transición de
Interfaz de Usuario (UITD)}
\author{Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.}
\date{}
\begin{document}
\maketitle
\section*{Afiliación}
Departamento de Matemáticas Aplicadas y Sistemas, Universidad Autónoma
Metropolitana, Ciudad de México, México
\section*{Contacto}
Contacto: jcervantes@.cua.uam.mx, dragonnomadal23@gmail.com,
mgomez@.cua.uam.mx

\section{Resumen}

El \textbf{Diagrama de Transición de Interfaz de Usuario (UITD)} es una
notación de modelado formal que simplifica la especificación y diseño
de interacciones entre usuarios y sistemas. En este artículo,
investigamos la eficiencia de crear UITDs utilizando \textit{draw.io},
un software de diagramación ampliamente utilizado, en comparación con
un editor UITD dedicado. Realizamos un estudio para comparar el tiempo
requerido para completar la tarea, así como la facilidad de uso
subjetiva y la satisfacción de los participantes. Nuestros resultados
muestran que el editor UITD es más eficiente y preferido por los
participantes, resaltando la importancia de utilizar herramientas
especializadas para crear modelos formales como los UITDs. Los
hallazgos de este estudio tienen implicaciones para desarrolladores de
software, diseñadores y otras partes interesadas involucradas en la
especificación y diseño de interacciones usuario-sistema.

\textbf{Palabras Clave:} Editor Eficiencia; Notación de Modelado;
Obtención de Requisitos

\section{Introducción}

El \textbf{UITD} es una notación de modelado formal que simplifica la
especificación y diseño de interacciones entre usuarios y sistemas. Se
puede utilizar para modelar el flujo de interfaces de usuario que
tendrá el sistema, convirtiéndose en una herramienta valiosa durante la
fase de obtención de requisitos. Además, es técnicamente preciso, lo
que permite su uso confiable para iniciar el desarrollo del sistema.

La simplicidad es clave al trabajar con notaciones de modelado,
especialmente para no expertos. El UITD proporciona una notación
```

completa y comprensible para las partes interesadas no técnicas, facilitando la comunicación entre diferentes grupos. En comparación con otras herramientas de modelado formal, el UITD tiene ventajas como su simplicidad, completitud y capacidad para representar de manera clara y concisa las interacciones usuario-sistema. Además, hay evidencia empírica sobre su comprensibilidad por parte de las partes no técnicas. Al usar el UITD, los diseñadores y desarrolladores pueden crear modelos técnicamente precisos y fáciles de entender, asegurando que todas las partes estén alineadas y satisfechas con la especificación de requisitos del sistema.

En los últimos años, se han desarrollado diversas herramientas para respaldar la creación de diagramas en general. Un ejemplo es `\textit{draw.io}`, un software de diagramación de código abierto ampliamente utilizado. Aunque `\textit{draw.io}` es versátil, puede que no esté específicamente diseñado para manejar los requisitos de creación de UITDs. Esto plantea la pregunta de cuán fácil es crear UITDs con `\textit{draw.io}` y si es más eficiente usar una herramienta dedicada como el editor UITD.

Para investigar esto, realizamos un estudio en el que los participantes crearon UITDs usando tanto `\textit{draw.io}` como el editor UITD. Nuestro estudio buscaba comparar la eficiencia de ambas herramientas...

`\end{document}`

### 3.2.11 Reescribir texto en Markdown en un archivo PDF

Solicita a CHATGPT que genere el código para escribir el artículo generado de Markdown en un archivo PDF

#### Solicitud:



Estoy usando Google Colab, genera el código Python para escribir el artículo generado en formato Markdown en un archivo PDF

#### Resultado:



Si estás utilizando Google Colab, puedes utilizar la biblioteca `reportlab` para generar un archivo PDF a partir del contenido Markdown. Aquí tienes el código que puedes ejecutar en Google Colab para lograrlo:

```
!pip install fpdf

from fpdf import FPDF
```

```
markdown_content = ""
```

```
# Ahorro de Tiempo con el Editor de Diagramas de Transición de  
Interfaz de Usuario (UITD)
```

```
*Autores*: Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.
```

```
*Afiliación*: Departamento de Matemáticas Aplicadas y Sistemas,  
Universidad Autónoma Metropolitana, Ciudad de México, México
```

```
*Contacto*: jcervantes@cua.uam.mx, dragonnomadal23@gmail.com,  
mgomez@cua.uam.mx
```

## ## Resumen

El **Diagrama de Transición de Interfaz de Usuario (UITD)** es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. En este artículo, investigamos la eficiencia de crear UITDs utilizando *draw.io*, un software de diagramación ampliamente utilizado, en comparación con un editor UITD dedicado. Realizamos un estudio para comparar el tiempo requerido para completar la tarea, así como la facilidad de uso subjetiva y la satisfacción de los participantes. Nuestros resultados muestran que el editor UITD es más eficiente y preferido por los participantes, resaltando la importancia de utilizar herramientas especializadas para crear modelos formales como los UITDs. Los hallazgos de este estudio tienen implicaciones para desarrolladores de software, diseñadores y otras partes interesadas involucradas en la especificación y diseño de interacciones usuario-sistema.

**Palabras Clave**: Editor Eficiencia; Notación de Modelado; Obtención de Requisitos

## ## Introducción

El **UITD** es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. Se puede utilizar para modelar el flujo de interfaces de usuario que tendrá el sistema, convirtiéndose en una herramienta valiosa durante la fase de obtención de requisitos. Además, es técnicamente preciso, lo que permite su uso confiable para iniciar el desarrollo del sistema.

La simplicidad es clave al trabajar con notaciones de modelado, especialmente para no expertos. El UITD proporciona una notación completa y comprensible para las partes interesadas no técnicas, facilitando la comunicación entre diferentes grupos. En comparación

con otras herramientas de modelado formal, el UITD tiene ventajas como su simplicidad, completitud y capacidad para representar de manera clara y concisa las interacciones usuario-sistema. Además, hay evidencia empírica sobre su comprensibilidad por parte de las partes no técnicas. Al usar el UITD, los diseñadores y desarrolladores pueden crear modelos técnicamente precisos y fáciles de entender, asegurando que todas las partes estén alineadas y satisfechas con la especificación de requisitos del sistema.

En los últimos años, se han desarrollado diversas herramientas para respaldar la creación de diagramas en general. Un ejemplo es *\*draw.io\**, un software de diagramación de código abierto ampliamente utilizado. Aunque *\*draw.io\** es versátil, puede que no esté específicamente diseñado para manejar los requisitos de creación de UITDs. Esto plantea la pregunta de cuán fácil es crear UITDs con *\*draw.io\** y si es más eficiente usar una herramienta dedicada como el editor UITD.

Para investigar esto, realizamos un estudio en el que los participantes crearon UITDs usando tanto *\*draw.io\** como el editor UITD. Nuestro estudio buscaba comparar la eficiencia de ambas herramientas...

"""

```
# Guardar el contenido Markdown en un archivo temporal
```

```
markdown_path = "/content/articulo1.md"
```

```
with open(markdown_path, "w") as markdown_file:
```

```
    markdown_file.write(markdown_content)
```

```
# Generar el archivo PDF
```

```
pdf = FPDF()
```

```
pdf.add_page()
```

```
pdf.set_auto_page_break(auto=True, margin=15)
```

```
pdf.set_font("Arial", size=12)
```

```
pdf.multi_cell(0, 10, markdown_content)
```

```
output_pdf_path = "/content/articulo1.pdf"
```

```
pdf.output(output_pdf_path)
```

```
print(f"El artículo se ha convertido a PDF y se ha guardado en:  
{output_pdf_path}")
```

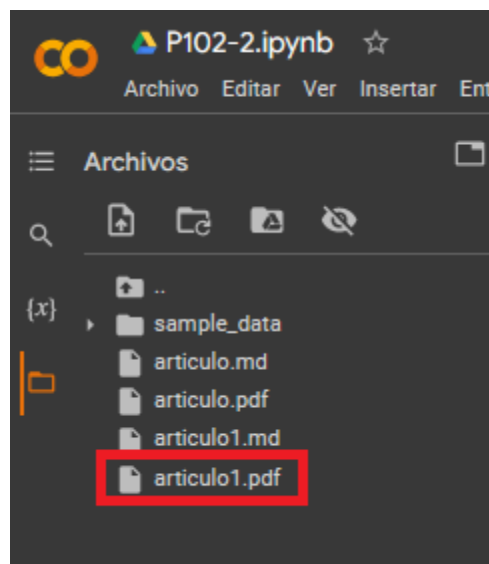
Este código instala la biblioteca `fpdf`, guarda el contenido Markdown en un archivo temporal, convierte el archivo Markdown en un archivo PDF utilizando `fpdf`, y finalmente muestra la ubicación del archivo PDF generado.



Resultado de ejecutar el código obtenido:

```
Collecting fpdf
  Downloading fpdf-1.7.2.tar.gz (39 kB)
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: fpdf
  Building wheel for fpdf (setup.py) ... done
  Created wheel for fpdf: filename=fpdf-1.7.2-py2.py3-none-any.whl size=40703
  sha256=9a08fda13c948c64b754594e478864614b92b038c8f7ebb8cb31f1def799efa5
  Stored in directory:
  /root/.cache/pip/wheels/f9/95/ba/f418094659025eb9611f17cbcaf2334236bf39a0c3453ea455
Successfully built fpdf
Installing collected packages: fpdf
Successfully installed fpdf-1.7.2
El artículo se ha convertido a PDF y se ha guardado en: /content/articulo1.pdf
```

Podemos observar que en la sección “Archivos” de Google Colab se ha generado el archivo con el nombre indicado en el código Python.



Abrimos el archivo `articulo1.pdf` y vemos un fragmento del contenido del archivo PDF:

# Ahorro de Tiempo con el Editor de Diagramas de Transición de Interfaz de Usuario (UITD)

\*Autores\*: Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.

\*Afiliación\*: Departamento de Matemáticas Aplicadas y Sistemas, Universidad Autónoma Metropolitana, Ciudad de México, México

\*Contacto\*: jcervantes@cua.uam.mx, dragonnomada123@gmail.com, mgomez@cua.uam.mx

## Resumen

El **Diagrama de Transición de Interfaz de Usuario (UITD)** es una notación de modelado formal que simplifica la especificación y diseño de interacciones entre usuarios y sistemas. En este artículo, investigamos la eficiencia de crear UITDs utilizando `draw.io`, un software de diagramación ampliamente utilizado, en comparación con un editor UITD dedicado. Realizamos un estudio para comparar el tiempo requerido para completar la tarea, así como la facilidad de uso subjetiva y la satisfacción de los participantes. Nuestros resultados muestran que el editor UITD es más eficiente y preferido por los participantes, resaltando la importancia de utilizar herramientas especializadas para crear modelos formales como los UITDs. Los hallazgos de este estudio tienen implicaciones para desarrolladores de software, diseñadores y otras partes interesadas involucradas en la especificación y diseño de interacciones usuario-sistema.

**Palabras Clave**: Editor Eficiencia; Notación de Modelado; Obtención de Requisitos

### **3.3 CASO DE ESTUDIO**

Diseña un caso de estudio dónde tomarías varias imágenes que contengan información relevante, por ejemplo, un acta constitutiva para extraer la empresa y la notaría que generaron el acta constitutiva y demás valores, un acta judicial dónde extraerías a personajes involucrados y causas, un estado de cuenta dónde extraerías montos y beneficiarios, etc.

#### **3.3.1 Diseño del caso de estudio**

El Maratón de la Ciudad de México es uno de los eventos deportivos más importantes y populares del país, atrayendo a corredores nacionales e internacionales de todas las edades y niveles. El evento se lleva a cabo anualmente y ofrece diferentes categorías, como el maratón completo (42.195 km) y un medio maratón para promover la participación masiva. El objetivo de este caso de estudio es analizar la convocatoria del Maratón de la Ciudad de México 2023, incluyendo su contenido y estructura con el fin de extraer información y poder hacer un análisis.

Queremos saber datos clave como:

- La fecha

- Número máximo de participantes

- Edades permitidas

- Si existe alguna indicación por la COVID-19

- En donde se realiza

- Los horarios de salida

- Costos

- etc.

La siguiente imagen contiene un fragmento de la convocatoria del Maratón CDMX 2023.



El Gobierno de la Ciudad de México, a través del Instituto del Deporte de la Ciudad de México, la Federación Mexicana de Asociaciones de Atletismo A.C. (FMAA), World Athletics (WA, por sus siglas en inglés) y la Asociación Internacional de Maratones y Carreras de Distancia (AIMS, por sus siglas en inglés).

### CONVOCAN

A las Federaciones Deportivas Nacionales e Internacionales, Asociaciones Estatales, Organismos, Instituciones, Clubes Deportivos Públicos y Privados, así como al público en general, a participar en el:

## “XL MARATÓN DE LA CIUDAD DE MÉXICO TELCEL 2023”

De conformidad con las siguientes:

### BASES

#### 1. FECHA

Domingo 27 de agosto de 2023.

#### 2. PARTICIPANTES

30,000 corredores

#### 3. LUGAR

Principales vialidades de la Ciudad de México, siguiendo una ruta de 42,195 metros, certificada y homologada por: FMAA, la AIMS y WA.

#### 4. SALIDA

Avenida Insurgentes Sur entre el edificio de la Biblioteca Central y Estadio Olímpico Universitario de la UNAM

##### Hora de Salida:

- Bloque de silla de ruedas y débiles visuales: arranque a partir de las 06:45 hrs.
- Bloque Elite Femenil 06:50 hrs.
- Bloque Elite Varonil 07:00 hrs.
- Bloque A: arranque a partir de las 07:00 hrs.
- Bloque B: arranque a partir de las 07:00 hrs.
- Bloque C: arranque a partir de las 07:00 hrs.

##### Meta:

Zócalo de la Ciudad de México

#### 5. REQUISITOS

- Edad mínima 18 años, cumplidos al día del evento.
- Corredores nacionales, cubrir el monto de inscripción.
- Corredores extranjeros, cubrir el monto de inscripción.

- Los corredores extranjeros ELITE deberán presentar el aval de su Federación respectiva y comprobar su calidad migratoria.
- Cada participante será responsable de verificar y apegarse a las medidas y recomendaciones sanitarias del Protocolo de Sanidad del XL Maratón de la Ciudad de México TELCEL 2023, mismo que podrá ser consultado en <https://indeporte.cdmx.gob.mx/>, <https://maraton.cdmx.gob.mx> y en medios oficiales del evento.
- En caso de que las autoridades sanitarias lo requieran, los corredores que hayan completado el esquema de vacunación deberán presentar el certificado que lo avale y el resultado negativo de una prueba COVID 19, con 72 horas previas como máximo. Aquellos con el esquema incompleto, deberán presentar una prueba COVID 19 NEGATIVA realizada con 72 horas previas como máximo.

#### 6. INSCRIPCIONES

Quedarán abiertas al público en general a partir del lanzamiento de la convocatoria y se cerrarán el miércoles 23 de agosto en Internet y Tiendas Martí; y el sábado 26 de agosto del 2023 a las 16:00 horas en la Expo Maratón, o antes en caso de que se agoten las inscripciones.

#### 7. COSTO

- **\$650 pesos** hasta el 31 de diciembre del 2022 y **\$700 pesos** a partir del 1 de enero 2023 - atletas nacionales a partir de la presente convocatoria.
- **\$100 USD** - atletas extranjeros a partir de la presente convocatoria.

Durante la Expo los días 24, 25 y 26 de agosto de 2023:

- **\$1,150 pesos** - todos los atletas (nacionales y extranjeros) durante la entrega de kits. Únicamente en apoyo a una de las fundaciones de “Corre con Causa” (en caso de disponibilidad

Observamos una imagen con texto relacionado con la convocatoria al Maratón de la Ciudad de México 2023. El encabezado de la imagen combina texto y patrones geométricos. Analizaremos si la extracción de ese texto se realiza con éxito.

### 3.3.2 Extracción de texto

Genera el código en Python (a través de CHATGPT) para extraer el texto con OCR y luego extraer la información con CHATGPT.

#### Solicitud:



Estoy usando Google Colab, dame el código Python usando OCR para extraer el texto de una imagen PNG que está guardada en Google Colab y se llama Caso\_Maraton.png

#### Resultado:



Por supuesto, puedes utilizar la biblioteca Tesseract para extraer el texto de una imagen en Google Colab. Asegúrate de que la imagen esté disponible en tu entorno de Colab antes de ejecutar el código siguiente. Aquí tienes un ejemplo de cómo hacerlo:

```
# Instalar Tesseract y su wrapper en Python (si no está instalado)
!pip install pytesseract
!apt-get install -y tesseract-ocr

# Importar las bibliotecas necesarias
import pytesseract
from PIL import Image
from google.colab import files

# Subir la imagen a Google Colab (si aún no está cargada)
uploaded = files.upload()

# Nombre del archivo de imagen cargado
image_filename = "Caso_Maraton.png" # Asegúrate de que el nombre sea correcto

# Cargar la imagen usando PIL
image = Image.open(image_filename)

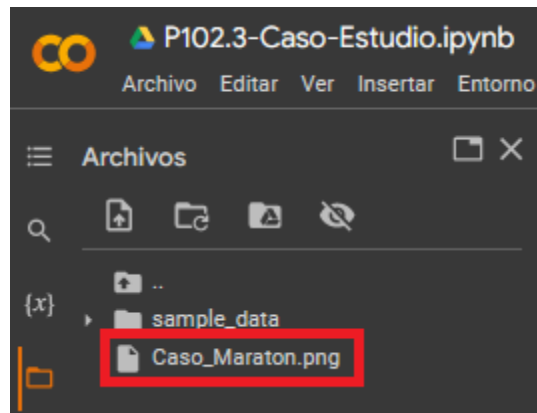
# Utilizar Tesseract para extraer el texto de la imagen
extracted_text = pytesseract.image_to_string(image)

# Imprimir el texto extraído
print(extracted_text)
```

Asegúrate de que la imagen "Caso\_Maraton.png" esté en la misma carpeta donde estás ejecutando el código o de proporcionar la ruta correcta hacia la imagen en el

código. Luego, ejecuta el código en una celda de Google Colab y debería imprimir el texto extraído de la imagen en la salida de la celda.

Hay que aclarar que posterior a la ejecución del código debimos colocar la imagen en la carpeta de trabajo en Google Colab.



Una vez que subimos la imagen y copiamos el código. Observamos que la extracción del texto se realizó de forma exitosa. Dividiremos los resultados en dos secciones.

LA primera donde se observan los resultados de Instalar Tesseract y su wrapper en Python:

```
Requirement already satisfied: pytesseract in /usr/local/lib/python3.10/dist-packages (0.3.10)
Requirement already satisfied: packaging>=21.3 in /usr/local/lib/python3.10/dist-packages (from pytesseract) (23.1)
Requirement already satisfied: Pillow>=8.0.0 in /usr/local/lib/python3.10/dist-packages (from pytesseract) (9.4.0)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
tesseract-ocr is already the newest version (4.1.1-2.1build1).
0 upgraded, 0 newly installed, 0 to remove and 16 not upgraded.
```

Y la segunda donde se imprime el texto extraído de la imagen:

```
Ct
CIUDAD
DE MEXICO
a (=) fa
```

El Gobierno de la Ciudad de México, a través del Instituto del Deporte de la Ciudad de México, la Federación Mexicana de Asociaciones de Atletismo A.C. (FMAA), World Athletics (WA, por sus siglas en inglés) y la Asociación Internacional de Maratones y Carreras de Distancia (AIMS, por sus siglas en inglés).

#### CONVOCAN

Alas Federaciones Deportivas Nacionales e Internacionales, Asociaciones Estatales, Organismos, Instituciones, Clubes Deportivos Públicos y Privados, así como al público en general, a participar en el:

"XL MARATON DE LA CIUDAD DE MEXICO TELCEL 2023"

De conformidad con las siguientes:

#### BASES

##### 1. FECHA

Domingo 27 de agosto de 2023.

##### 2. PARTICIPANTES

30,000 corredores

##### 3. LUGAR

Principales vialidades de la Ciudad de México, siguiendo una ruta de 42,195 metros, certificada y homologada por: FMAA, la AIMS y WA.

##### 4. SALIDA

Avenida Insurgentes Sur entre el edificio de la Biblioteca Central y Estadio Olímpico Universitario de la UNAM

Hora de Salida:

+ Bloque de silla de ruedas y débiles visuales: arranque a partir de las 06:45 hrs.

+ Bloque Elite Femenil 06:50 hrs.

+ Bloque Elite Varonil 07:00 hrs.

+ Bloque A: arranque a partir de las 07:00 hrs.

+ Bloque B: arranque a partir de las 0 hrs.

+ Bloque C: arranque a partir de las 07:00 hrs.

Meta:

Zocalo de la Ciudad de México

##### 5. REQUISITOS

- + Edad mínima 18 años, cumplidos al día del evento.
- + Corredores nacionales, cubrir el monto de inscripción.
- + Corredores extranjeros, cubrir el monto de inscripción.

+ Los corredores extranjeros ELITE deberán presentar el aval de su Federación respectiva y comprobar su calidad migratoria.

+ Cada participante será responsable de verificar y apegarse a las medidas y recomendaciones sanitarias del Protocolo de Sanidad del XL Maratón de la Ciudad de México TELCEL 2023, mismo que podrá ser consultado en <https://indeporte.cdmx.gob.mx/>, <https://maraton.cdmx.gob.mx> y en medios oficiales del evento.

+ En caso de que las autoridades sanitarias lo requieran, los corredores que hayan completado el esquema de vacunación deberán presentar el certificado que lo avale y el resultado negativo de una prueba COVID 19, con 72 horas previas como máximo. Aquellos con el esquema incompleto, deberán presentar una prueba COVID 19 NEGATIVA realizada con 72 horas previas como máximo.

#### 6. INSCRIPCIONES

Quedarán abiertas al público en general a partir del lanzamiento de la convocatoria y se cerrarán el miércoles 23 de agosto en Internet y Tiendas Martí; y el sábado 26 de agosto del 2023 a las 16:00 horas en la Expo Maratón, o antes en caso de que se agoten las inscripciones.

#### 7. COSTO

+ \$650 pesos hasta el 31 de diciembre del 2022 y \$700 pesos a partir del 1 de enero 2023 - atletas nacionales a partir de la presente convocatoria.

+ \$100 USD - atletas extranjeros a partir de la presente convocatoria.

Durante la Expo los días 24, 25 y 26 de agosto de 2023:

+» \$1,150 pesos - todos los atletas (nacionales y extranjeros) durante la entrega de kits. Únicamente en apoyo a una de las fundaciones de "Corre con Causa" (en caso de disponibilidad

Se observa que el texto del encabezado, el cual está entrelazado con patrones geométricos, no fue exitosamente extraído, a excepción del fragmento "Ciudad de México". Por otro lado, la totalidad de la información restante fue extraída de manera precisa.

### 3.3.3 Análisis y extracción de información del texto

Ahora introduzcamos el texto obtenido a ChatGPT para ver que nos dice la IA.

#### Solicitud:

CI

Te daré el siguiente texto:

"Ct CIUDAD DE MEXICO a (=) fa El Gobierno de la Ciudad de México, a través del Instituto del Deporte de la Ciudad de México, la Federación Mexicana de Asociaciones de Atletismo A.C. (FMAA), World Athletics (WA, por sus siglas en inglés) y la Asociación



Internacional de Maratones y Carreras de Distancia (AIMS, por sus siglas en inglés). CONVOCAN Alas Federaciones Deportivas Nacionales e Internacionales, Asociaciones Estatales, Organismos, Instituciones, Clubes Deportivos Ptblicos y Privados, asi como al ptblico en general, a participar en el: "XL MARATON DE LA CIUDAD DE MEXICO TELCEL 2023" De conformidad con las siguientes: BASES 1. FECHA Domingo 27 de agosto de 2023. 2. PARTICIPANTES 30,000 corredores 3. LUGAR Principales vialidades de la Ciudad de México, siguiendo una ruta de 42,195 metros, certificada y homologada por: FMAA, la AIMS y WA. 4. SALIDA Avenida Insurgentes Sur entre el edificio de la Biblioteca Central y Estadio Olimpico Universitario de la UNAM Hora de Salida: + Bloque de silla de ruedas y débiles visuales: arranque a partir de las 06:45 hrs. + Bloque Elite Femenil 06:50 hrs. + Bloque Elite Varonil 07:00 hrs. + Bloque A: arranque a partir de las 07:00 hrs. + Bloque B: arranque a partir de las 0 hrs. + Bloque C: arranque a partir de las 07:00 hrs. Meta: Zocalo de la Ciudad de México 5. REQUISITOS + Edad minima 18 afios, cumplidos al dia del evento. + Corredores nacionales, cubrir el monto de inscripcié + Corredores extranjeros, cubrir el monto de inscripcién. + Los corredores extranjeros ELITE deberan presentar el aval de su Federacién respectiva y comprobar su calidad migratoria. + Cada participante sera responsable de verificar y apegarse a las medidas y recomendaciones sanitarias del Protocolo de Sanidad del XL Maratén de la Ciudad de México TELCEL 2023, mismo que podra ser consultado en <https://indeporte.cdmx.gob.mx/>, <https://maraton.cdmx.gob.mx> y en medios oficiales del evento. + En caso de que las autoridades sanitarias lo requieran, los corredores que hayan completado el esquema de vacunacion deberdn presentar el certificado que lo avale y el resultado negativo de una prueba COVID 19, con 72 horas previas como maximo. Aquellos con el esquema incompleto, deberdn presentar una prueba COVID 19 NEGATIVA realizada con 72 horas previas como maximo. 6. INSCRIPCIONES Quedaran abiertas al ptblico en general a partir del lanzamiento de la convocatoria y se cerraran el miércoles 23 de agosto en Internet y Tiendas Marti; y el sabado 26 de agosto del 2023 a las 16:00 horas en la Expo Maraton, 0 antes en caso de que se agoten las inscripciones. 7. COSTO + \$650 pesos hasta el 31 de diciembre del 2022 y \$700 pesos a partir del 1 de enero 2023 - atletas nacionales a partir de la presente convocatoria. + \$100 USD - atletas extranjeros a partir de la presente convocatoria. Durante la Expo los dias 24, 25 y 26 de agosto de 2023: +» \$1,150 pesos - todos los atletas (nacionales y extranjeros) durante la entrega de kits. Unicamente en apoyo a una de las fundaciones de "Corre con Causa" (en caso de disponibilidad"

## Resultado:



Parece que me has proporcionado un fragmento de texto que parece ser una convocatoria para un maratón en la Ciudad de México en el año 2023. ¿En qué puedo ayudarte con este texto? ¿Necesitas algún tipo de resumen, aclaración o información específica?

Al introducir el texto extraído de la imagen y proporcionarla al ChatGPT vemos que la IA detecta que se trata de un fragmento de la convocatoria del Maratón CDMX 2023. Probemos extraer alguna información

#### Solicitud:

CI

Quiénes son los sujetos involucrados en el texto?

#### Resultado:



Los sujetos involucrados en el texto son:

1. **El Gobierno de la Ciudad de México:** Organización gubernamental responsable de la administración de la ciudad.
2. **Instituto del Deporte de la Ciudad de México:** Entidad encargada de promover y organizar actividades deportivas en la ciudad.
3. **Federación Mexicana de Asociaciones de Atletismo A.C. (FMAA):** Organización que regula y promueve el atletismo en México.
4. **World Athletics (WA):** Organización internacional que supervisa y regula el atletismo a nivel global.
5. **Asociación Internacional de Maratones y Carreras de Distancia (AIMS):** Organización que regula y promueve maratones y carreras de distancia en todo el mundo.
6. **Federaciones Deportivas Nacionales e Internacionales, Asociaciones Estatales, Organismos, Instituciones, Clubes Deportivos Públicos y Privados:** Diversas entidades deportivas a nivel nacional e internacional.
7. **Público en general:** Personas interesadas en participar en el evento.
8. **Corredores:** Participantes que se inscriben para correr en el maratón.
9. **Aval de su Federación respectiva:** Documento que respalda la calidad y categoría deportiva de los corredores extranjeros élite.
10. **Autoridades sanitarias:** Entidades encargadas de la salud pública y regulación sanitaria.
11. **Fundaciones de "Corre con Causa":** Organizaciones benéficas que pueden recibir apoyo a través de las inscripciones al evento.

Cada uno de estos sujetos tiene un papel específico en la organización, promoción y desarrollo del "XL Maratón de la Ciudad de México TELCEL 2023".

#### Solicitud:

CI

Del texto anterior dame la siguiente información:  
Fecha del Maratón  
Número de corredores

Cuando comienzan las inscripciones  
Cuando terminan las inscripciones  
Edad mínima permitida  
¿Existe alguna restricción por COVID-19?  
Costos de inscripción  
Cantidad de bloques de salida  
Tipos de bloques de salida  
Lugar de Salida  
Lugar de la Meta

## Resultado:



Claro, aquí tienes la información solicitada basada en el texto proporcionado:

- **Fecha del Maratón:** Domingo 27 de agosto de 2023.
- **Número de corredores:** Se permiten un total de 30,000 corredores.
- **Inicio de inscripciones:** Las inscripciones quedaron abiertas al público en general a partir del lanzamiento de la convocatoria.
- **Fin de inscripciones:** Las inscripciones se cerrarán el miércoles 23 de agosto en Internet y Tiendas Marti, y el sábado 26 de agosto de 2023 a las 16:00 horas en la Expo Maratón, o antes en caso de que se agoten las inscripciones.
- **Edad mínima permitida:** Los participantes deben tener un mínimo de 18 años cumplidos al día del evento.
- **Restricciones por COVID-19:** Los participantes deben apegarse a las medidas y recomendaciones sanitarias del Protocolo de Sanidad del XL Maratón de la Ciudad de México TELCEL 2023. En caso de que las autoridades sanitarias lo requieran, los corredores deben presentar el certificado de vacunación completo y un resultado negativo de una prueba COVID-19 realizada en las 72 horas previas como máximo.
- **Costos de inscripción:**
  - \$650 pesos hasta el 31 de diciembre del 2022 y \$700 pesos a partir del 1 de enero 2023 para atletas nacionales.
  - \$100 USD para atletas extranjeros.

Durante la Expo los días 24, 25 y 26 de agosto de 2023, el costo será de \$1,150 pesos para todos los atletas (nacionales y extranjeros) durante la entrega de kits, en apoyo a una de las fundaciones "Corre con Causa" (en caso de disponibilidad).
- **Cantidad de bloques de salida:** Hay varios bloques de salida para diferentes categorías de corredores.
- **Tipos de bloques de salida:**
  - Bloque de silla de ruedas y débiles visuales: A partir de las 06:45 hrs.
  - Bloque Elite Femenil: 06:50 hrs.
  - Bloque Elite Varonil: 07:00 hrs.
  - Bloque A: A partir de las 07:00 hrs.
  - Bloque B: A partir de las 07:00 hrs.
  - Bloque C: A partir de las 07:00 hrs.
- **Lugar de Salida:** Avenida Insurgentes Sur entre el edificio de la Biblioteca Central y el Estadio Olímpico Universitario de la UNAM.
- **Lugar de la Meta:** Zócalo de la Ciudad de México.

Recuerda que la información se basa en el fragmento de texto que proporcionaste y podría haber más detalles en el texto completo.

### Solicitud:

CI

Del texto que te proporcione genera un resumen

### Resultado:



El Gobierno de la Ciudad de México, en colaboración con diversas entidades deportivas nacionales e internacionales, convoca a participar en el "XL Maratón de la Ciudad de México TELCEL 2023". El evento se llevará a cabo el 27 de agosto de 2023, con un límite de 30,000 corredores. El recorrido de 42,195 metros por las principales vialidades de la ciudad ha sido certificado por las federaciones deportivas. Las inscripciones están abiertas desde el lanzamiento de la convocatoria y se cerrarán el 23 de agosto en línea y en tiendas físicas, o antes si se agotan. La edad mínima para participar es de 18 años. Se deben seguir medidas sanitarias debido a la COVID-19, incluyendo la presentación de certificados de vacunación completos y resultados negativos de pruebas recientes. Los costos varían según la fecha y la nacionalidad, y durante la Expo Maratón, los atletas pueden inscribirse en apoyo a fundaciones benéficas. Habrá varios bloques de salida para diferentes categorías de corredores, y la carrera iniciará en la Avenida Insurgentes Sur y culminará en el Zócalo de la Ciudad de México.

### Solicitud:

CI

Genera el código Markdown de tabla que contenga los bloques de salida en una columna y en otra columna la hora de salida

### Resultado:



Por supuesto, aquí está el código Markdown para la tabla con los bloques de salida y las horas correspondientes:

Bloque de Salida	Hora de Salida
Silla de ruedas y débiles visuales	06:45 hrs
Elite Femenil	06:50 hrs
Elite Varonil	07:00 hrs
Bloque A	07:00 hrs
Bloque B	07:00 hrs
Bloque C	07:00 hrs

Puedes copiar y pegar este código Markdown en cualquier editor compatible para visualizar la tabla.

Solicitud:



Unifica la lista de información que te solicite, el resumen y la tabla en un solo texto

Resultado:

CI

Unifica la lista de información que te solicite, el resumen y la tabla en un solo texto

Por supuesto, aquí tienes la información unificada en un solo texto, que incluye el resumen, los detalles clave y la tabla con los bloques de salida y horas correspondientes:

### XL Maratón de la Ciudad de México TELCEL 2023

El Gobierno de la Ciudad de México, en colaboración con el Instituto del Deporte de la Ciudad de México, la Federación Mexicana de Asociaciones de Atletismo A.C. (FMAA), World Athletics (WA) y la Asociación Internacional de Maratones y Carreras de Distancia (AIMS), te invita a participar en el **XL Maratón de la Ciudad de México TELCEL 2023**. Este emocionante evento deportivo se llevará a cabo el 27 de agosto de 2023, en las principales vialidades de la ciudad.

#### Detalles clave:

- **Fecha del Maratón:** Domingo 27 de agosto de 2023.
- **Número de Corredores:** Se permiten un máximo de 30,000 participantes.
- **Inicio de Inscripciones:** Desde el lanzamiento de la convocatoria.
- **Fin de Inscripciones:** Cierre el miércoles 23 de agosto en línea y en tiendas físicas, o antes si se agotan.
- **Edad Mínima:** Participantes deben tener al menos 18 años cumplidos al día del evento.

#### Restricciones por COVID-19:

Se deben seguir medidas sanitarias debido a la COVID-19, incluyendo la presentación de certificados de vacunación completos y resultados negativos de pruebas recientes.

#### Costos de Inscripción:

- **Atletas nacionales:** \$650 pesos hasta el 31 de diciembre del 2022, \$700 pesos a partir del 1 de enero de 2023.
- **Atletas extranjeros:** \$100 USD.

Durante la Expo Maratón, los días 24, 25 y 26 de agosto de 2023, los atletas pueden inscribirse con un costo de \$1,150 pesos en apoyo a fundaciones benéficas.

#### Bloques de Salida y Horas:

Bloque de Salida	Hora de Salida
Silla de ruedas y débiles visuales	06:45 hrs
Elite Femenil	06:50 hrs
Elite Varonil	07:00 hrs
Bloque A	07:00 hrs
Bloque B	07:00 hrs
Bloque C	07:00 hrs

#### Recorrido:

La carrera iniciará en la Avenida Insurgentes Sur, entre el edificio de la Biblioteca Central y el Estadio Olímpico Universitario de la UNAM, y culminará en el Zócalo de la Ciudad de México.

Para más detalles y consultas sobre las medidas sanitarias, puedes visitar el sitio web oficial: <https://indeporte.cdmx.gob.mx/>

¡Únete a este emocionante evento y vive la experiencia del XL Maratón de la Ciudad de México TELCEL 2023!

### 3.3.4 Creación del Reporte

#### Solicitud:



Dame el código para presentar esta última información en formato Markdown

#### Resultado:



Por supuesto, aquí tienes el código Markdown para presentar toda la información unificada:

```
# XL Maratón de la Ciudad de México TELCEL 2023

El Gobierno de la Ciudad de México, en colaboración con el Instituto del Deporte de la Ciudad de México, la Federación Mexicana de Asociaciones de Atletismo A.C. (FMAA), World Athletics (WA) y la Asociación Internacional de Maratones y Carreras de Distancia (AIMS), te invita a participar en el XL Maratón de la Ciudad de México TELCEL 2023. Este emocionante evento deportivo se llevará a cabo el 27 de agosto de 2023, en las principales vialidades de la ciudad.

## Detalles clave:

- Fecha del Maratón: Domingo 27 de agosto de 2023.
- Número de Corredores: Se permiten un máximo de 30,000 participantes.
- Inicio de Inscripciones: Desde el lanzamiento de la convocatoria.
- Fin de Inscripciones: Cierre el miércoles 23 de agosto en línea y en tiendas físicas, o antes si se agotan.
- Edad Mínima: Participantes deben tener al menos 18 años cumplidos al día del evento.

## Restricciones por COVID-19:

Se deben seguir medidas sanitarias debido a la COVID-19, incluyendo la presentación de certificados de vacunación completos y resultados negativos de pruebas recientes.

## Costos de Inscripción:

- Atletas nacionales: $650 pesos hasta el 31 de diciembre del 2022, $700 pesos a partir del 1 de enero de 2023.
- Atletas extranjeros: $100 USD.
```

Durante la Expo Maratón, los días 24, 25 y 26 de agosto de 2023, los atletas pueden inscribirse con un costo de \$1,150 pesos en apoyo a fundaciones benéficas.

### ## Bloques de Salida y Horas:

Bloque de Salida	Hora de Salida
-----	-----
Silla de ruedas y débiles visuales	06:45 hrs
Elite Femenil	06:50 hrs
Elite Varonil	07:00 hrs
Bloque A	07:00 hrs
Bloque B	07:00 hrs
Bloque C	07:00 hrs

### ## Recorrido:

La carrera iniciará en la Avenida Insurgentes Sur, entre el edificio de la Biblioteca Central y el Estadio Olímpico Universitario de la UNAM, y culminará en el Zócalo de la Ciudad de México.


Para más detalles y consultas sobre las medidas sanitarias, puedes visitar el sitio web oficial:

[<https://indeporte.cdmx.gob.mx/>] (<https://indeporte.cdmx.gob.mx/>)

¡Únete a este emocionante evento y vive la experiencia del XL Maratón de la Ciudad de México TELCEL 2023!

Este código Markdown se puede copiar y pegar en un editor compatible para mostrar la información formateada correctamente. Este código Markdown se puede copiar y pegar en un editor compatible para mostrar la información formateada correctamente.

Si introducimos este código Markdown a Google Colab obtenemos el siguiente resultado:



## XL Maratón de la Ciudad de México TELCEL 2023

El Gobierno de la Ciudad de México, en colaboración con el Instituto del Deporte de la Ciudad de México, la Federación Mexicana de Asociaciones de Atletismo A.C. (FMAA), World Athletics (WA) y la Asociación Internacional de Maratones y Carreras de Distancia (AIMS), te invita a participar en el **XL Maratón de la Ciudad de México TELCEL 2023**. Este emocionante evento deportivo se llevará a cabo el 27 de agosto de 2023, en las principales vialidades de la ciudad.

### Detalles clave:

- **Fecha del Maratón:** Domingo 27 de agosto de 2023.
- **Número de Corredores:** Se permiten un máximo de 30,000 participantes.
- **Inicio de Inscripciones:** Desde el lanzamiento de la convocatoria.
- **Fin de Inscripciones:** Cierre el miércoles 23 de agosto en línea y en tiendas físicas, o antes si se agotan.
- **Edad Mínima:** Participantes deben tener al menos 18 años cumplidos al día del evento.

### Restricciones por COVID-19:

Se deben seguir medidas sanitarias debido a la COVID-19, incluyendo la presentación de certificados de vacunación completos y resultados negativos de pruebas recientes.

### Costos de Inscripción:

- **Atletas nacionales:** 650 pesos hasta el 31 de diciembre de 2022, 700 pesos a partir del 1 de enero de 2023.
- **Atletas extranjeros:** \$100 USD.

Durante la Expo Maratón, los días 24, 25 y 26 de agosto de 2023, los atletas pueden inscribirse con un costo de \$1,150 pesos en apoyo a fundaciones benéficas.

### Bloques de Salida y Horas:

Bloque de Salida	Hora de Salida
Silla de ruedas y débiles visuales	06:45 hrs
Elite Femenil	06:50 hrs
Elite Varonil	07:00 hrs
Bloque A	07:00 hrs
Bloque B	07:00 hrs
Bloque C	07:00 hrs

### Recorrido:

La carrera iniciará en la Avenida Insurgentes Sur, entre el edificio de la Biblioteca Central y el Estadio Olímpico Universitario de la UNAM, y culminará en el Zócalo de la Ciudad de México.

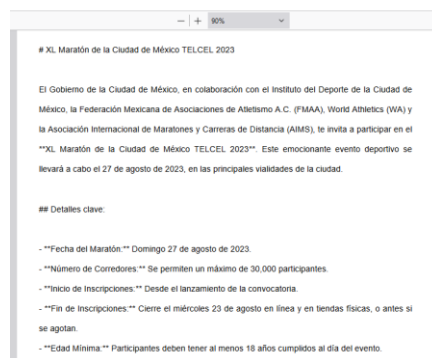
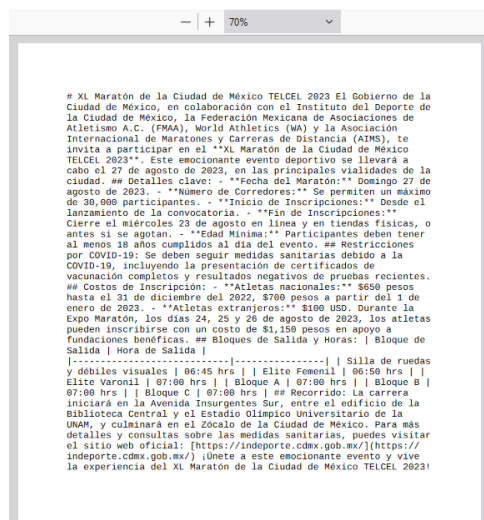
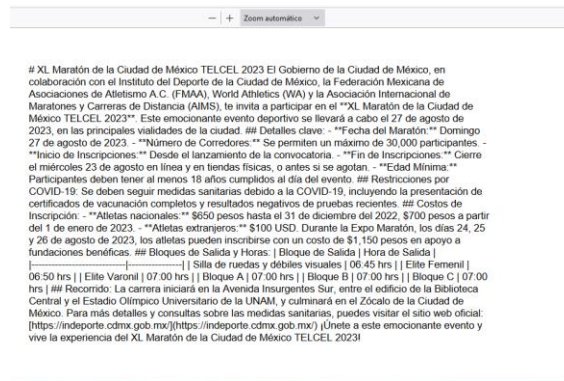
Para más detalles y consultas sobre las medidas sanitarias, puedes visitar el sitio web oficial:  
<https://indeporte.cdmx.gob.mx/>

¡Únete a este emocionante evento y vive la experiencia del XL Maratón de la Ciudad de México TELCEL 2023!



No logré generar el PDF exitosamente utilizando el código Markdown interpretado.

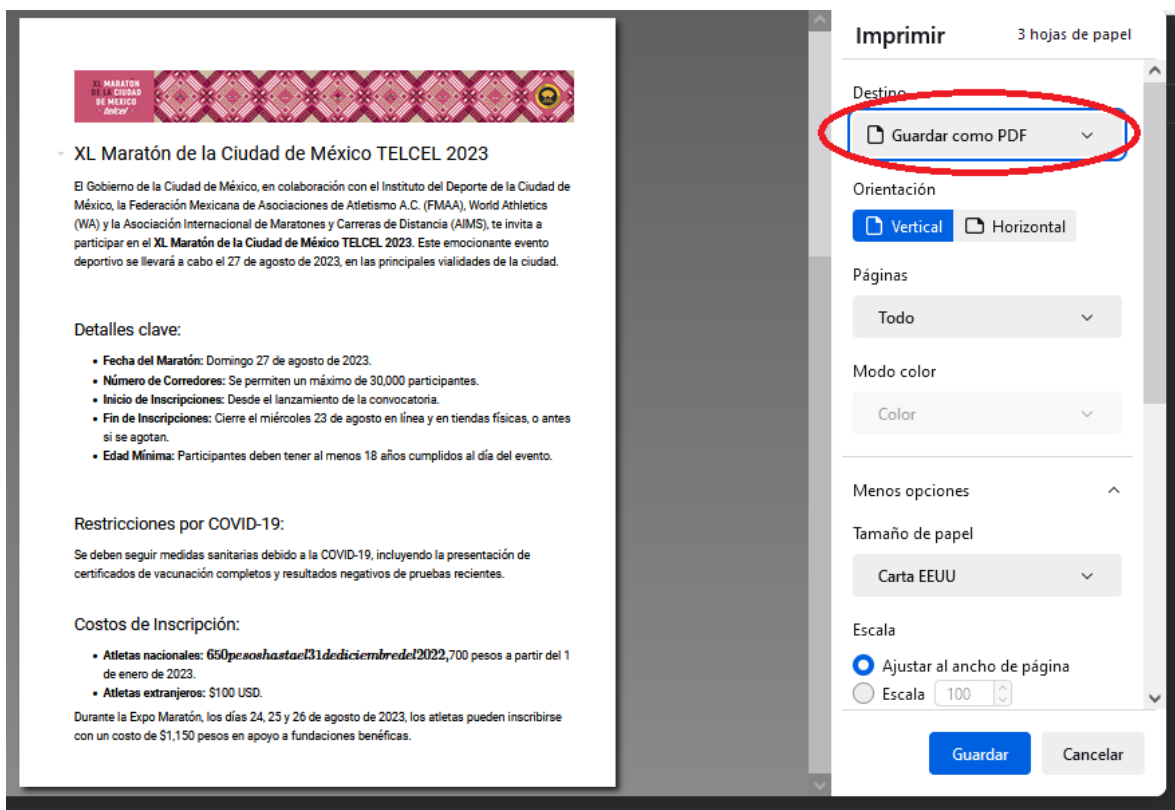
Aunque ChatGPT me proporcionó varias alternativas para crear el archivo PDF, los resultados obtenidos no cumplían con mis expectativas. Mis diversos intentos resultaron en un PDF que contenía elementos como los siguientes:



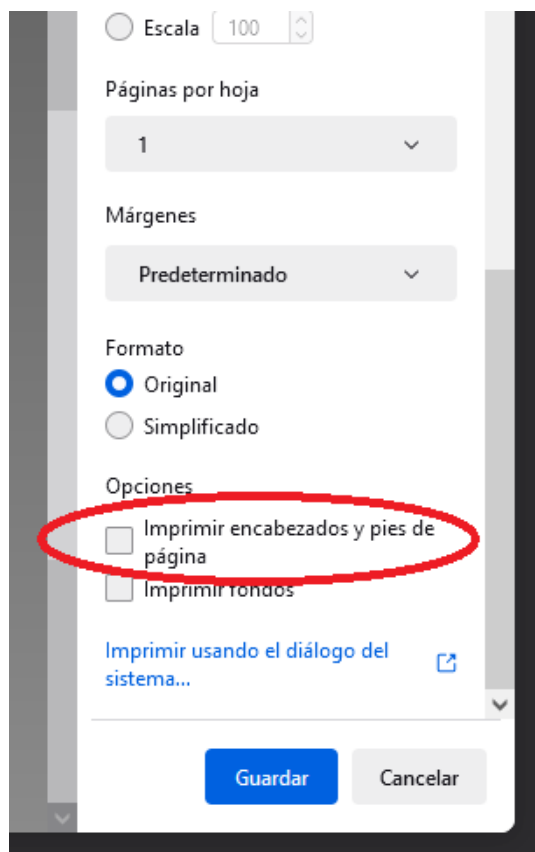
Esperaba obtener el resultado visto en Google Colab pero en un archivo PDF. Tras varios intentos encontré que desde Google Colab se puede generar un PDF. Así que procedí a abrir un nuevo cuaderno, pegué el código Markdown, guardarlo y después para crear el PDF hay que ir al menú Archivo de Google Colab y elegir la opción Imprimir.



y después en la sección Destino seleccionar la opción Guardar como PDF:



Yo recomiendo desactivar la opción “Imprimir encabezados y pies de página” para que la presentación sea mejor.




Incluí una imagen representativa del Maratón como encabezado, con el propósito de mejorar la presentación del informe en formato PDF. Para insertar imágenes en una celda de texto en Google Colab, se emplea la siguiente sintaxis:

```
![Texto alternativo](URL_de_la_imagen)
```

Esto permite la incorporación de la imagen alojada en la URL proporcionada.

Así el resultado final el PDF es el siguiente:



## XL Maratón de la Ciudad de México TELCEL 2023

El Gobierno de la Ciudad de México, en colaboración con el Instituto del Deporte de la Ciudad de México, la Federación Mexicana de Asociaciones de Atletismo A.C. (FMAA), World Athletics (WA) y la Asociación Internacional de Maratones y Carreras de Distancia (AIMS), te invita a participar en el **XL Maratón de la Ciudad de México TELCEL 2023**. Este emocionante evento deportivo se llevará a cabo el 27 de agosto de 2023, en las principales vialidades de la ciudad.

**Detalles clave:**

- **Fecha del Maratón:** Domingo 27 de agosto de 2023.
- **Número de Corredores:** Se permiten un máximo de 30,000 participantes.
- **Inicio de Inscripciones:** Desde el lanzamiento de la convocatoria.
- **Fin de Inscripciones:** Cierre el miércoles 23 de agosto en línea y en tiendas físicas, o antes si se agotan.
- **Edad Mínima:** Participantes deben tener al menos 18 años cumplidos al día del evento.

**Restricciones por COVID-19:**

Se deben seguir medidas sanitarias debido a la COVID-19, incluyendo la presentación de certificados de vacunación completos y resultados negativos de pruebas recientes.

**Costos de Inscripción:**

- **Atletas nacionales:** 650 pesos hasta el 31 de diciembre del 2022, 700 pesos a partir del 1 de enero de 2023.
- **Atletas extranjeros:** \$100 USD.

Durante la Expo Maratón, los días 24, 25 y 26 de agosto de 2023, los atletas pueden inscribirse con un costo de \$1,150 pesos en apoyo a fundaciones benéficas.

### Bloques de Salida y Horas:

Bloque de Salida	Hora de Salida
Silla de ruedas y débiles visuales	06:45 hrs
Elite Femenil	06:50 hrs
Elite Varonil	07:00 hrs
Bloque A	07:00 hrs
Bloque B	07:00 hrs
Bloque C	07:00 hrs

**Recorrido:**

La carrera iniciará en la Avenida Insurgentes Sur, entre el edificio de la Biblioteca Central y el Estadio Olímpico Universitario de la UNAM, y culminará en el Zócalo de la Ciudad de México.

Para más detalles y consultas sobre las medidas sanitarias, puedes visitar el sitio web oficial: <https://indeporte.cdmx.gob.mx/>

¡Únete a este emocionante evento y vive la experiencia del XL Maratón de la Ciudad de México TELCEL 2023!

Haz doble clic (o pulsa Intro) para editar

Queda pendiente la creación del PDF a través de Python obteniendo este mismo resultado, que contenga la interpretación de Markdown generada por ChatGPT. Seguiré trabajando en ello.

## 4. CONCLUSIONES

Cuando se dispone de un conjunto de datos almacenados en un archivo CSV, su utilidad por sí sola es limitada. Sin embargo, al emplear herramientas de análisis, se transforma en una valiosa fuente de información. A partir de este análisis, es posible obtener conclusiones específicas o tomar decisiones fundamentadas.

Con ChatGPT, es posible generar código en Python para realizar análisis y visualizaciones a partir de conjuntos de datos simplemente expresándolo en lenguaje natural. La fusión de estas dos tecnologías resulta especialmente beneficiosa para abordar desafíos de esta índole, incluso en ausencia de un conocimiento profundo en programación. ChatGPT produce código funcional en la mayoría de los casos, y en las pocas ocasiones en que no es efectivo, es posible brindar información sobre la plataforma utilizada, como Google Colab, o describir los errores generados por el código, permitiendo que la IA realice correcciones exitosas.

Aunque en mi experiencia personal, en raras ocasiones el código no ha operado como esperaba, en esos casos he logrado indicarle a ChatGPT que estoy utilizando Google Colab o he detallado los errores del código, obteniendo soluciones satisfactorias. A pesar de que en esta ocasión no conseguí generar el archivo PDF de la manera deseada, pude demostrar la capacidad de Google Colab para crear archivos PDF con el formato definido en Markdown. Esto representa una alternativa válida en ciertos contextos. Al revisar las prácticas de mis compañeros en el repositorio del profesor en GitHub, noté que algunos de ellos lograron alcanzar ese objetivo. Planeo emplear sus ejemplos como referencia e intentar replicarlos, continuando mi trabajo en esta dirección. También noté la posibilidad de usar `pandoc` para generar el PDF, lo cual exploraré.

Cuando se trata de extraer información de imágenes, es asombroso cómo Python puede reconocer y extraer texto. Sin embargo, cuando el texto está integrado con imágenes u

otros patrones, la extracción puede no ser tan precisa, tal como se evidenció en el caso de estudio que diseñé y desarrollé. Una vez que logramos extraer el texto, pudimos analizarlo con ChatGPT para obtener información específica y presentarla en formato Markdown, facilitando la creación del archivo PDF.

El dominio y manejo de diversas herramientas resultan esenciales para alcanzar nuestros objetivos. Existen múltiples formas de abordar tareas, incluyendo ChatGPT, Google Colab, pandoc, entre otras. Estas herramientas demostraron su utilidad en el transcurso de esta práctica.