



INSTITUTO POLITÉCNICO NACIONAL

CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN

INTRODUCCIÓN A CHATGPT CON PYTHON

ALUMNO:
ORLANDO DAVID LOPEZ ESPINOSA

PROFESOR:
ALAN BADILLO SALAS

PRÁCTICA 102

AGOSTO 2023

P102.1 - Reporte estadístico

GN

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



¡Por supuesto! Puedes usar la biblioteca Pandas en Python para abrir un archivo CSV desde una URL. Aquí tienes un ejemplo de cómo puedes hacerlo para el archivo `weight-height.csv` desde la URL proporcionada:

python

Copy code

```
import pandas as pd

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

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

Este código importa la biblioteca Pandas, define la URL del archivo CSV, y luego utiliza la función `pd.read_csv()` para leer el archivo desde la URL. Finalmente, imprime las primeras filas del DataFrame resultante para verificar que los datos se hayan cargado correctamente.

```
import pandas as pd

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

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

	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

GN

Ahora genera el código que imprima 10 valores aleatorios del dataset anterior usando la función `sample` de Pandas



¡Claro! La función `sample()` de Pandas es muy útil para obtener un subconjunto aleatorio de filas de un DataFrame. Aquí te muestro cómo obtener 10 valores aleatorios del dataset:

python

Copy code

```
import pandas as pd

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Obtener 10 filas aleatorias del DataFrame
df_aleatorio = df.sample(n=10)

# Imprimir las 10 filas aleatorias
print(df_aleatorio)
```

La función `sample()` se encarga de seleccionar las filas al azar sin reemplazo. Si quisieras, por ejemplo, seleccionar una fracción del total de filas (por ejemplo, el 10% de las filas) podrías usar el argumento `frac` en lugar de `n`. Por ejemplo, `df.sample(frac=0.1)` seleccionaría aleatoriamente el 10% de las filas del DataFrame.



```
import pandas as pd

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Obtener 10 filas aleatorias del DataFrame
df_aleatorio = df.sample(n=10)

# Imprimir las 10 filas aleatorias
print(df_aleatorio)
```

	Gender	Height	Weight
9714	Female	60.884873	128.530333
5300	Female	57.397740	106.587563
3512	Male	69.759638	196.731372
6148	Female	64.842595	142.879180
2832	Male	67.500067	175.879804
4047	Male	69.039522	192.092449
4541	Male	67.525973	175.470021
2808	Male	69.693408	201.481472
8846	Female	60.876915	130.068216
763	Male	64.408273	154.641042

GN

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



Si deseas obtener una descripción de la columna categórica `Gender` y contar las ocurrencias de cada valor único, puedes usar las funciones `unique()` y `value_counts()` de Pandas. Aquí está el código:

python

Copy code

```
import pandas as pd

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Extraer los valores únicos de la columna 'Gender'
valores_unicos = df['Gender'].unique()

# Contar las ocurrencias de cada valor único en la columna 'Gender'
conteo_valores = df['Gender'].value_counts()

# Imprimir los resultados
print("Valores únicos en la columna 'Gender':")
print(valores_unicos)
print("\nConteo de cada valor en la columna 'Gender':")
print(conteo_valores)
```

- `unique()` devuelve un array con los valores únicos en el orden en que aparecen en el DataFrame.
- `value_counts()` devuelve una serie con el conteo de cada valor único. Por defecto, los conteos se ordenan de mayor a menor.

```
import pandas as pd

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Extraer los valores únicos de la columna 'Gender'
valores_unicos = df['Gender'].unique()

# Contar las ocurrencias de cada valor único en la columna 'Gender'
conteo_valores = df['Gender'].value_counts()

# Imprimir los resultados
print("Valores únicos en la columna 'Gender':")
print(valores_unicos)
print("\nConteo de cada valor en la columna 'Gender':")
print(conteo_valores)
```

Valores únicos en la columna 'Gender':
['Male' 'Female']

Conteo de cada valor en la columna 'Gender':
Male 5000
Female 5000
Name: Gender, dtype: int64

GN

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



Por supuesto, aquí tienes un ejemplo de código que calcula las estadísticas solicitadas para la columna `Height` en el archivo CSV proporcionado:

python

Copy code

```
import pandas as pd
import scipy.stats as stats

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Extraer la columna 'Height'
altura = df['Height']

# Calcular estadísticas
total = altura.count()
minimo = altura.min()
maximo = altura.max()
promedio = altura.mean()
desviacion_estandar = altura.std()
intervalo_confianza = stats.t.interval(alpha=0.95, df=len(altura)-1, loc=promedio,
                                       scale=desviacion_estandar,
                                       quantiles=[0.25, 0.5, 0.75])
cuartiles = altura.quantile([0.25, 0.5, 0.75])

# Imprimir los resultados
print("Estadísticas para la columna 'Height':")
print("Total:", total)
print("Mínimo:", minimo)
print("Máximo:", maximo)
print("Promedio:", promedio)
print("Desviación Estándar:", desviacion_estandar)
print("Intervalo de Confianza al 95%:", intervalo_confianza)
print("Cuartiles:")
print(cuartiles)
```

- `count()`, `min()`, `max()`, `mean()`, y `std()` son funciones de Pandas que calculan las estadísticas básicas.
- La función `interval()` de la biblioteca SciPy se utiliza para calcular el intervalo de confianza.
- La función `quantile()` devuelve los cuartiles solicitados (en este caso, los percentiles 25, 50 y 75).

Asegúrate de tener instaladas las bibliotecas Pandas y SciPy para ejecutar este código.


```

import pandas as pd
import scipy.stats as stats

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Extraer la columna 'Height'
altura = df['Height']

# Calcular estadísticas
total = altura.count()
minimo = altura.min()
maximo = altura.max()
promedio = altura.mean()
desviacion_estandar = altura.std()
intervalo_confianza = stats.t.interval(alpha=0.95, df=len(altura)-1, loc=promedio, scale=desviacion_estandar)
cuartiles = altura.quantile([0.25, 0.5, 0.75])

# Imprimir los resultados
print("Estadísticas para la columna 'Height':")
print("Total:", total)
print("Mínimo:", minimo)
print("Máximo:", maximo)
print("Promedio:", promedio)
print("Desviación Estándar:", desviacion_estandar)
print("Intervalo de Confianza al 95%:", intervalo_confianza)
print("Cuartiles:")
print(cuartiles)

```

```

Estadísticas para la columna 'Height':
Total: 10000
Mínimo: 54.2631333250971
Máximo: 78.9987423463896
Promedio: 66.36755975482124
Desviación Estándar: 3.8475281207732293
Intervalo de Confianza al 95%: (58.825630270662856, 73.90948923897963)
Cuartiles:
0.25    63.585620
0.50    66.318070
0.75    69.174262
Name: Height, dtype: float64

```

```

<ipython-input-4-e645cc4f3e33>:19: DeprecationWarning: Use of keyword argument 'alpha' for method 'interval' is deprecated and will be removed in SciPy 1.11.0. Use first positional argument or keyword argument 'confidence' instead.
intervalo_confianza = stats.t.interval(alpha=0.95, df=len(altura)-1, loc=promedio, scale=desviacion_estandar)

```

GN

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



Por supuesto. Vamos a calcular las estadísticas solicitadas para la columna `Weight`:



python

Copy code

```
import pandas as pd
import scipy.stats as stats

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Extraer la columna 'Weight'
peso = df['Weight']

# Calcular estadísticas
total = peso.count()
minimo = peso.min()
maximo = peso.max()
promedio = peso.mean()
desviacion_estandar = peso.std()
intervalo_confianza = stats.t.interval(alpha=0.95, df=len(peso)-1, loc=promedio,
                                       scale=desviacion_estandar,
                                       lower=0, upper=None)
cuartiles = peso.quantile([0.25, 0.5, 0.75])

# Imprimir los resultados
print("Estadísticas para la columna 'Weight':")
print("Total:", total)
print("Mínimo:", minimo)
print("Máximo:", maximo)
print("Promedio:", promedio)
print("Desviación Estándar:", desviacion_estandar)
print("Intervalo de Confianza al 95%:", intervalo_confianza)
print("Cuartiles:")
print(cuartiles)
```

Este código es similar al anterior pero centrado en la columna `Weight`. Utiliza funciones de Pandas para calcular estadísticas básicas y la función `interval()` de SciPy para obtener el intervalo de confianza del 95%.

```

import pandas as pd
import scipy.stats as stats

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Extraer la columna 'Weight'
peso = df['Weight']

# Calcular estadísticas
total = peso.count()
minimo = peso.min()
maximo = peso.max()
promedio = peso.mean()
desviacion_estandar = peso.std()
intervalo_confianza = stats.t.interval(alpha=0.95, df=len(peso)-1, loc=promedio, scale=desviacion_estandar)
cuartiles = peso.quantile([0.25, 0.5, 0.75])

# Imprimir los resultados
print("Estadísticas para la columna 'Weight':")
print("Total:", total)
print("Mínimo:", minimo)
print("Máximo:", maximo)
print("Promedio:", promedio)
print("Desviación Estándar:", desviacion_estandar)
print("Intervalo de Confianza al 95%:", intervalo_confianza)
print("Cuartiles:")
print(cuartiles)

```

```

Estadísticas para la columna 'Weight':
Total: 10000
Mínimo: 64.700126712753
Máximo: 269.989698505106
Promedio: 161.44035683283076
Desviación Estándar: 32.108439006519596
Intervalo de Confianza al 95%: (98.50135411934176, 224.37935954631976)
Cuartiles:
0.25    135.818051
0.50    161.212928
0.75    187.169525
Name: Weight, dtype: float64
<ipython-input-5-b1045ca18ae0>:19: DeprecationWarning: Use of keyword argument 'alpha' for method 'interval' is deprecated
  intervalo_confianza = stats.t.interval(alpha=0.95, df=len(peso)-1, loc=promedio, scale=desviacion_estandar)

```



¡Por supuesto! Aquí tienes un ejemplo de cómo crear un histograma para la columna `Height` usando la biblioteca Seaborn:

Primero, necesitarás instalar las bibliotecas pertinentes si aún no lo has hecho:

diff

Copy code

```
!pip install pandas seaborn
```

Ahora, el código para graficar el histograma:

python

Copy code

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

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Configurar el estilo y tamaño del gráfico
sns.set(style="whitegrid")
plt.figure(figsize=(10,6))

# Crear el histograma usando Seaborn
sns.histplot(df['Height'], kde=True, bins=30)

# Dar título y etiquetas al gráfico
plt.title('Histograma de la columna Height')
plt.xlabel('Height')
plt.ylabel('Frecuencia')

# Mostrar el gráfico
plt.show()
```

En este código:

- Usamos `sns.histplot` para graficar el histograma.
- El argumento `kde=True` añade una estimación de densidad del kernel (una línea suavizada que da una idea general de la distribución).
- `bins=30` especifica que queremos 30 barras en el histograma. Puedes ajustar este número según tus preferencias o necesidades.

```

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

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

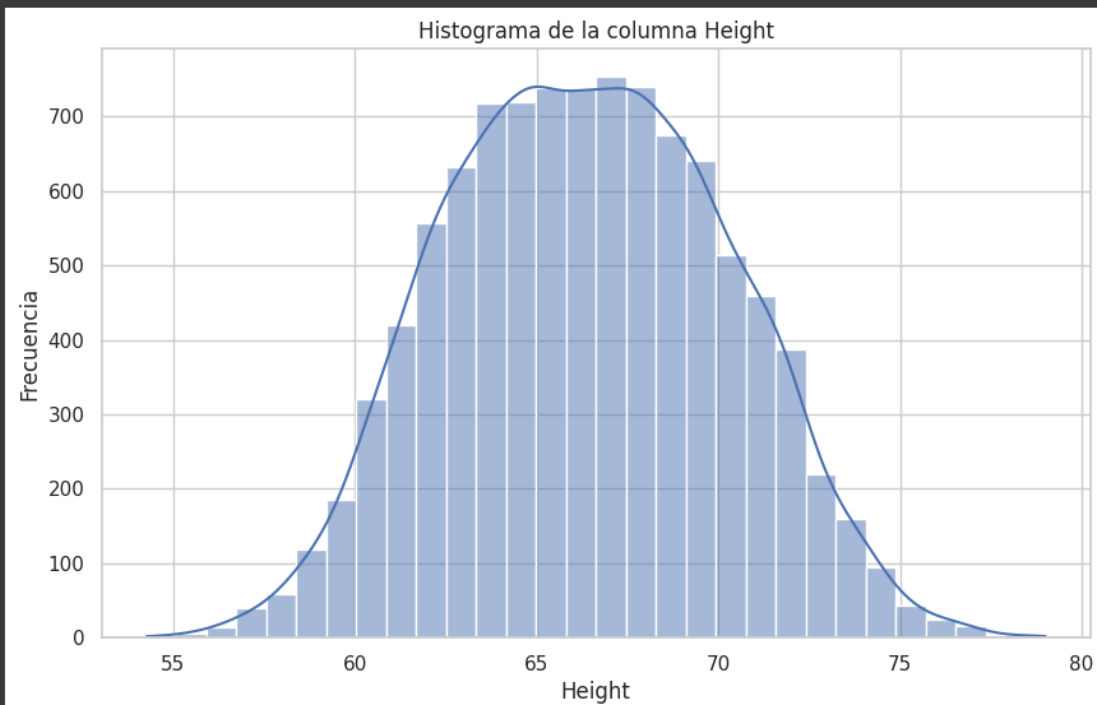
# Configurar el estilo y tamaño del gráfico
sns.set(style="whitegrid")
plt.figure(figsize=(10,6))

# Crear el histograma usando Seaborn
sns.histplot(df['Height'], kde=True, bins=30)

# Dar título y etiquetas al gráfico
plt.title('Histograma de la columna Height')
plt.xlabel('Height')
plt.ylabel('Frecuencia')

# Mostrar el gráfico
plt.show()

```





Genera el código que grafique el histograma de la columna Weight usando Seaborn



¡Claro! A continuación, te muestro cómo graficar un histograma para la columna `Weight`

usando Seaborn:

python

Copy code

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

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Configurar el estilo y tamaño del gráfico
sns.set(style="whitegrid")
plt.figure(figsize=(10,6))

# Crear el histograma usando Seaborn
sns.histplot(df['Weight'], kde=True, bins=30)

# Dar título y etiquetas al gráfico
plt.title('Histograma de la columna Weight')
plt.xlabel('Weight')
plt.ylabel('Frecuencia')

# Mostrar el gráfico
plt.show()
```

Este código es muy similar al anterior pero centrándose en la columna `Weight`. Se utiliza `sns.histplot` para crear el histograma, y se añade la estimación de densidad del kernel con `kde=True` para tener una representación suavizada de la distribución de datos.

```

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

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

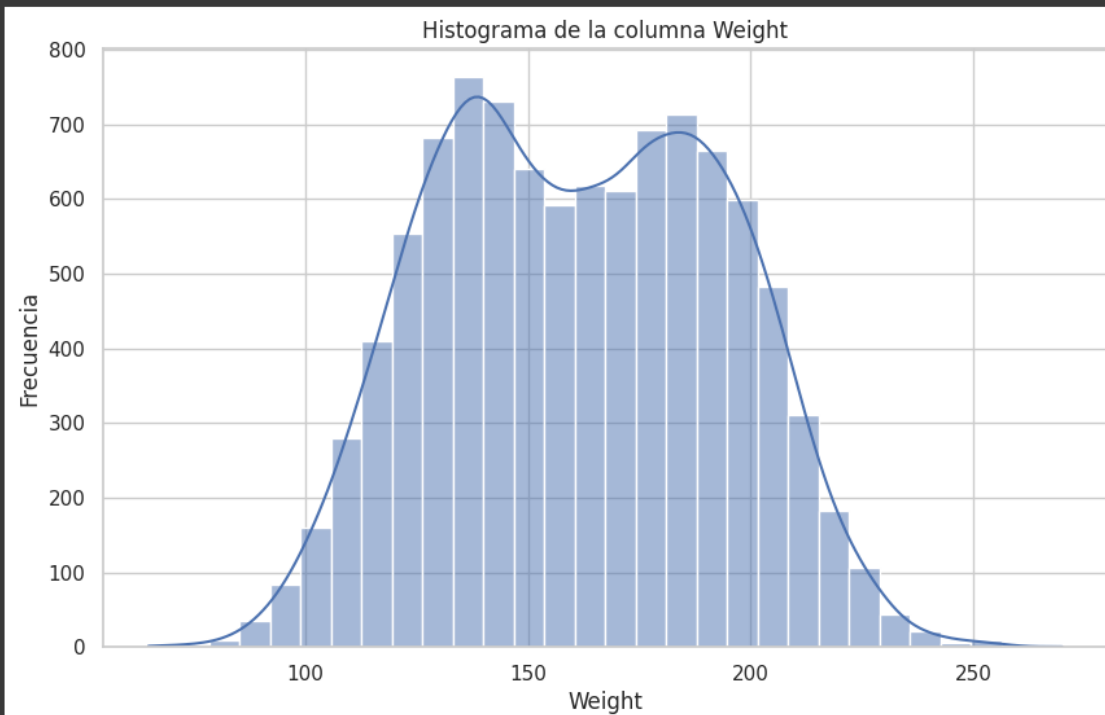
# Configurar el estilo y tamaño del gráfico
sns.set(style="whitegrid")
plt.figure(figsize=(10,6))

# Crear el histograma usando Seaborn
sns.histplot(df['Weight'], kde=True, bins=30)

# Dar título y etiquetas al gráfico
plt.title('Histograma de la columna Weight')
plt.xlabel('Weight')
plt.ylabel('Frecuencia')

# Mostrar el gráfico
plt.show()

```



GN

Genera el código que grafique la correlación entre las columnas Height y Weight usando Seaborn



Para visualizar la correlación entre las columnas `Height` y `Weight`, uno de los gráficos más utilizados es el scatter plot o gráfico de dispersión. Usando Seaborn, puedes combinar este gráfico con una línea de regresión para obtener una representación visual de la relación entre estas dos variables.

A continuación, te muestro cómo hacerlo:

python

Copy code

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

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Configurar el estilo y tamaño del gráfico
sns.set(style="whitegrid")
plt.figure(figsize=(10,6))

# Gráfico de dispersión con línea de regresión
sns.regplot(x=df['Height'], y=df['Weight'], scatter_kws={"s": 10}, line_kws={"color": "red"})

# Dar título y etiquetas al gráfico
plt.title('Correlación entre Height y Weight')
plt.xlabel('Height')
plt.ylabel('Weight')

# Mostrar el gráfico
plt.show()
```

En este código:

- Usamos `sns.regplot` para crear el gráfico de dispersión y la línea de regresión.
- `scatter_kws` y `line_kws` son argumentos para personalizar el aspecto de los puntos y la línea de regresión, respectivamente. En este caso, ajustamos el tamaño de los puntos con `{"s": 10}` y el color de la línea de regresión con `{"color": "red"}`.

Este gráfico te permitirá visualizar cómo se relacionan las variables `Height` y `Weight` y qué tan fuerte es esa relación.


```

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

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

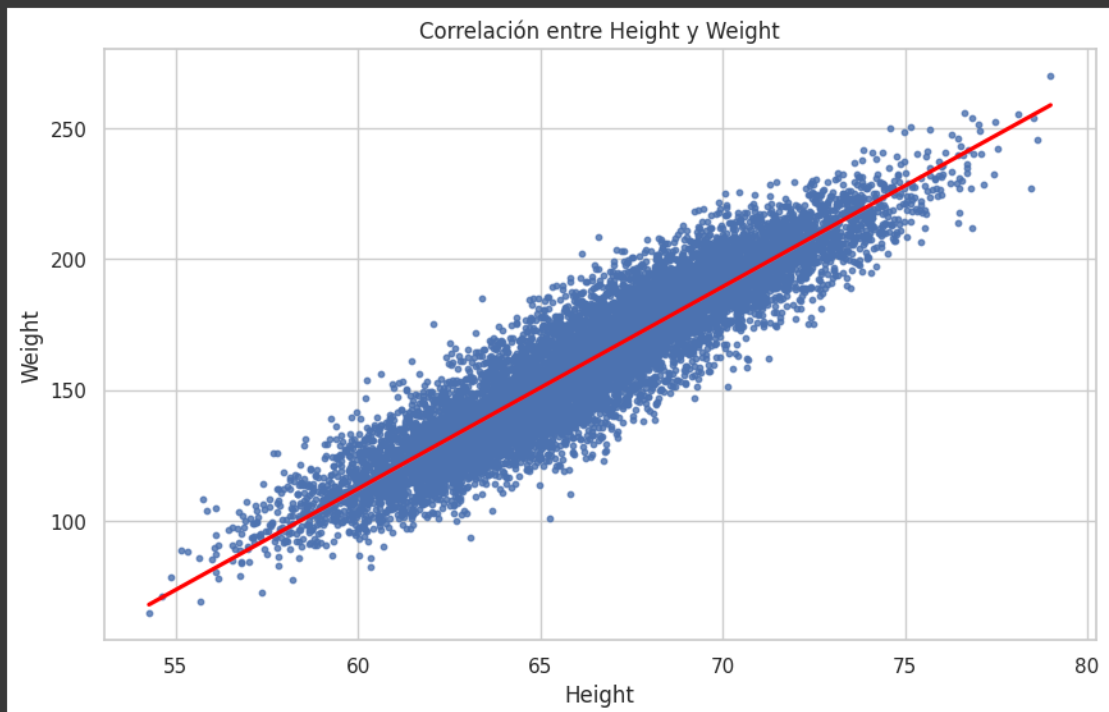
# Configurar el estilo y tamaño del gráfico
sns.set(style="whitegrid")
plt.figure(figsize=(10,6))

# Gráfico de dispersión con línea de regresión
sns.regplot(x=df['Height'], y=df['Weight'], scatter_kws={"s": 10}, line_kws={"color": "red"})

# Dar título y etiquetas al gráfico
plt.title('Correlación entre Height y Weight')
plt.xlabel('Height')
plt.ylabel('Weight')

# Mostrar el gráfico
plt.show()

```





Genera el código que obtenga la correlación entre las columnas Height y Weight usando corr de Numpy



Para calcular la correlación entre dos series de datos en Python, puedes usar la función `corrcoef()` de Numpy. Esta función devuelve una matriz de correlaciones, y el valor que estamos buscando es la entrada fuera de la diagonal.

A continuación te muestro cómo obtener la correlación entre las columnas `Height` y `Weight` usando `corrcoef()`:

python

Copy code

```
import pandas as pd
import numpy as np

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Obtener la correlación usando numpy
correlation_matrix = np.corrcoef(df['Height'], df['Weight'])
correlation = correlation_matrix[0, 1]

print(f"Correlación entre Height y Weight: {correlation:.2f}")
```

El valor de correlación resultante oscilará entre -1 y 1. Un valor cercano a 1 indica una correlación positiva fuerte, un valor cercano a -1 indica una correlación negativa fuerte y un valor cercano a 0 indica poca o ninguna correlación.



```
import pandas as pd
import numpy as np

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Obtener la correlación usando numpy
correlation_matrix = np.corrcoef(df['Height'], df['Weight'])
correlation = correlation_matrix[0, 1]

print(f"Correlación entre Height y Weight: {correlation:.2f}")
```

Correlación entre Height y Weight: 0.92

GN

Genera el código 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



¡Entendido! Aquí tienes un código que calcula y reporta los valores promedio de las columnas `Height` y `Weight` para cada género:

python

Copy code

```
import pandas as pd

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Calcular el promedio de Height y Weight para hombres
avg_height_male = df[df['Gender'] == 'Male']['Height'].mean()
avg_weight_male = df[df['Gender'] == 'Male']['Weight'].mean()

# Calcular el promedio de Height y Weight para mujeres
avg_height_female = df[df['Gender'] == 'Female']['Height'].mean()
avg_weight_female = df[df['Gender'] == 'Female']['Weight'].mean()

# Imprimir los resultados
print(f"Promedio de Height para hombres: {avg_height_male:.2f}")
print(f"Promedio de Weight para hombres: {avg_weight_male:.2f}")
print(f"Promedio de Height para mujeres: {avg_height_female:.2f}")
print(f"Promedio de Weight para mujeres: {avg_weight_female:.2f}")
```

Este código utiliza filtros de Pandas para separar los datos por género y luego calcula el promedio (`mean()`) de las columnas requeridas.

```
import pandas as pd

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

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Calcular el promedio de Height y Weight para hombres
avg_height_male = df[df['Gender'] == 'Male']['Height'].mean()
avg_weight_male = df[df['Gender'] == 'Male']['Weight'].mean()

# Calcular el promedio de Height y Weight para mujeres
avg_height_female = df[df['Gender'] == 'Female']['Height'].mean()
avg_weight_female = df[df['Gender'] == 'Female']['Weight'].mean()

# Imprimir los resultados
print(f"Promedio de Height para hombres: {avg_height_male:.2f}")
print(f"Promedio de Weight para hombres: {avg_weight_male:.2f}")
print(f"Promedio de Height para mujeres: {avg_height_female:.2f}")
print(f"Promedio de Weight para mujeres: {avg_weight_female:.2f}")
```

```
Promedio de Height para hombres: 69.03
Promedio de Weight para hombres: 187.02
Promedio de Height para mujeres: 63.71
Promedio de Weight para mujeres: 135.86
```

P102.2 - Extracción de Información



Genera el código que extraiga el texto de una imagen PNG desde la URL https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/UITD_PAPER.png usando OCR en Python



Para extraer texto de una imagen a través de una URL utilizando OCR en Python, puedes hacer uso de la librería **pytesseract** y **PIL** (parte de **Pillow**). Además, necesitarás la librería **requests** para descargar la imagen desde la URL. Aquí te muestro cómo hacerlo paso a paso:

1. Instala las librerías necesarias:

Copy code

```
pip install pytesseract Pillow requests
```

Nota: También necesitarás tener instalado Tesseract en tu máquina. Si estás usando un sistema basado en Debian/Ubuntu, puedes instalarlo con:

Copy code

```
sudo apt-get install tesseract-ocr
```

1. Código Python para descargar la imagen y extraer texto:

Copy code

```
python

import requests
from PIL import Image
from io import BytesIO
import pytesseract

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

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

# Extrae el texto usando pytesseract
text = pytesseract.image_to_string(image)

print(text)
```

Cuando ejecutes este script, debería mostrarte el texto extraído de la imagen ubicada en la URL proporcionada.

Si la calidad de la imagen no es óptima o tiene un fondo ruidoso, es posible que necesites realizar algún preprocesamiento en la imagen (como binarización, eliminación de ruido, etc.) para mejorar la precisión del OCR.



El error que estás viendo indica que el ejecutable de **`tesseract`** no se encuentra en el PATH de tu sistema o no está instalado correctamente. Aquí hay algunas soluciones según tu sistema operativo:

1. Si estás en Linux (Debian/Ubuntu):

Instala Tesseract usando el administrador de paquetes:

bash

Copy code

```
sudo apt-get update
sudo apt-get install tesseract-ocr
```

2. Si estás en macOS:

Puedes usar Homebrew para instalar Tesseract:

bash

Copy code

```
brew install tesseract
```

3. Si estás en Windows:

- Puedes descargar el instalador de Tesseract desde [este enlace](#).
- Una vez instalado, debes asegurarte de que el ejecutable de Tesseract esté en tu PATH o especificar la ruta completa al ejecutable en tu script de Python. Por ejemplo, si instalaste Tesseract en **`C:\Program Files\Tesseract-OCR`**, puedes agregar la siguiente línea a tu script:

python

Copy code

```
pytesseract.pytesseract.tesseract_cmd = r'C:\Program Files\Tesseract-OCR\tesseract.exe'
```

Luego de seguir estos pasos según tu sistema operativo, vuelve a intentar ejecutar tu script. Debería poder encontrar el ejecutable de Tesseract y procesar la imagen correctamente.



Si estás trabajando en Google Colab, el proceso de instalación y configuración de Tesseract es un poco diferente. Aquí te muestro cómo hacerlo paso a paso:



1. Instalar Tesseract en Google Colab:

Ejecuta la siguiente celda en tu notebook de Colab para instalar Tesseract:

python

Copy code

```
!sudo apt install tesseract-ocr
```

1. Instalar las librerías de Python necesarias:

Aún en Google Colab, instala las librerías requeridas:

python

Copy code

```
!pip install pytesseract Pillow requests
```

1. Escribe el código para extraer el texto:

Después de instalar todo, puedes utilizar el código que proporcioné anteriormente para descargar la imagen y extraer texto de ella:

python

Copy code

```
import requests
from PIL import Image
from io import BytesIO
import pytesseract

# Dirección de la imagen
URL = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/data"

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

# Extrae el texto usando pytesseract
text = pytesseract.image_to_string(image)

print(text)
```

Simplemente ejecuta las celdas en el orden indicado y deberías obtener el texto extraído de la imagen. No es necesario configurar el PATH ni nada similar en Colab, ya que Tesseract se

```
[ ] import requests
    from PIL import Image
    from io import BytesIO
    import pytesseract

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

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

    # Extrae el texto usando pytesseract
    text = pytesseract.image_to_string(image)

    print(text)
```

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

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

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



Aquí tienes las traducciones solicitadas:



1. "Ahorro de Tiempo con el Editor de Diagramas de Transición de Interfaz de Usuario (UITD)"
2. "El Editor UITD: Una Herramienta para Simplificar el Modelado de Interacción Usuario-Sistema"
3. "Optimizando el Diseño de la Interfaz de Usuario con el Editor UITD: Un Estudio Comparativo"
4. "Editor UITD: Simplificando el Modelado de Interacción Usuario-Sistema para Desarrolladores"

¿El editor de Diagramas de Transición de Interfaz de Usuario (editor UITD) puede 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, 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 usuario-sistema. Es una valiosa herramienta de comunicación para las partes interesadas técnicas y no técnicas durante la fase de obtención de requisitos, ya que proporciona una notación sencilla y completa que es fácil de entender. En este documento, investigamos la eficiencia de crear UITDs usando 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, destacando la importancia de usar herramientas especializadas para crear modelos formales como 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.

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 diseño de interacciones usuario-sistema. Puede ser utilizado para modelar el flujo de interfaces de usuario que el sistema tendrá, convirtiéndolo en una valiosa herramienta de comunicación para las partes interesadas técnicas y no técnicas durante la fase de obtención de requisitos. Además, es técnicamente preciso, lo que permite utilizarlo de manera fiable para iniciar el desarrollo del sistema [1].

Regenerate

La simplicidad es clave al trabajar con notaciones de modelado, particularmente para no expertos [3]. El UITD proporciona una notación que es completa y simple lo suficiente para que las partes interesadas no técnicas la entiendan [2]. Es una notación de modelado formal que es fácil de aprender, facilitando la comunicación entre las partes interesadas con y sin habilidades técnicas de desarrollo de software. Comparado con otras herramientas de modelado formal, el UITD tiene varias ventajas, incluida su simplicidad, completitud y capacidad para modelar interacciones usuario-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, diseñadores y desarrolladores pueden crear modelos que son técnicamente precisos y fáciles de entender, ayudando a asegurar que todas las partes interesadas estén alineadas y satisfechas con la especificación de los requisitos del sistema.

En años recientes, se han desarrollado varias herramientas para apoyar la creación de diagramas generalizados. Una de estas herramientas es draw.io, que es un software de diagramación conocido y de código abierto. Draw.io es ampliamente utilizado para crear diagramas, flujogramas y otros tipos de representaciones gráficas. Aunque draw.io es una herramienta versátil, es posible que no esté específicamente diseñado 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 buscaba comparar la eficiencia de las dos herramientas en.

(Note: El texto termina en "en", lo que indica que probablemente haya más contenido que no se proporcionó en el texto original).



Quiénes son los autores del artículo



Los autores del artículo son:



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

Están afiliados al Departamento de Matemáticas Aplicadas y Sistemas de la Universidad Autónoma Metropolitana en Ciudad de México, México.



Cuáles son los correos relacionados al artículo



Los correos electrónicos relacionados al artículo son:



- jcervantes@cua.uam.mx
- dragonnomadal23@gmail.com
- mgomez@cua.uam.mx



Cuál es la universidad, el departamento, la ciudad y el país del artículo



De acuerdo al artículo, la información es:



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



Claro, desglosaré el abstract para darte una explicación más clara:

**Tema Principal:**

El documento trata sobre el Diagrama de Transición de Interfaz de Usuario (UITD), una notación de modelado formal diseñada para simplificar la especificación y diseño de las interacciones entre el usuario y el sistema.

Importancia:

- El UITD sirve como herramienta de comunicación entre diferentes partes interesadas, tanto técnicas como no técnicas, especialmente durante la fase de obtención de requisitos.
- Provee una notación que es a la vez completa y sencilla, facilitando su comprensión.

Objetivo del Estudio:

Los autores investigaron la eficacia de la herramienta draw.io (un software popular para la creación de diagramas) en comparación con un editor específico para UITD.

Método:

Realizaron un estudio donde compararon el tiempo que se tarda en completar tareas usando ambas herramientas, así como la facilidad de uso percibida y la satisfacción de los participantes.

Resultados:

Descubrieron que el editor específico para UITD es más eficiente y fue preferido por los participantes, subrayando la relevancia de utilizar herramientas especializadas para la creación de modelos formales como el UITD.

Implicaciones:

Este descubrimiento es relevante para los desarrolladores de software, diseñadores y otros interesados que participan en la especificación y diseño de interacciones entre el usuario y el sistema, sugiriendo que herramientas especializadas pueden ser más beneficiosas para tareas específicas.

En resumen, el abstract presenta un estudio comparativo sobre dos herramientas para modelar interacciones usuario-sistema. Se encontró que una herramienta especializada para UITD es más efectiva que una herramienta de diagramación general como draw.io.

Regenerate



Explicame la introducción



Por supuesto, a continuación te ofrezco un desglose simplificado de la introducción:



Tema Principal:

- La introducción se centra en el Diagrama de Transición de Interfaz de Usuario (UITD), una notación de modelado formal. Su principal propósito es simplificar la especificación y diseño de las interacciones entre el usuario y el sistema.

Utilidad del UITD:

- Sirve para modelar el flujo de las interfaces de usuario que tendrá un sistema.
- Actúa como una herramienta de comunicación esencial entre las partes interesadas, ya sean técnicas o no técnicas, durante la fase de obtención de requisitos.
- Aporta precisión técnica, lo que significa que es fiable y puede ser utilizado para iniciar el desarrollo real del sistema.

Características del UITD:

- La simplicidad es fundamental cuando se trata de notaciones de modelado, especialmente para aquellos que no son expertos en el área.
- El UITD es fácil de aprender y proporciona una notación completa, pero también es lo suficientemente sencillo para que las personas sin habilidades técnicas en desarrollo de software lo entiendan.
- Comparado con otras herramientas de modelado, el UITD tiene ventajas como su simplicidad, completitud y claridad para modelar interacciones usuario-sistema.

Evidencia Empírica:

- Se menciona que existen evidencias empíricas que demuestran que el UITD es comprensible incluso para las partes interesadas que no son técnicas.
- Utilizar el UITD ayuda a que los modelos creados sean técnicamente precisos y fáciles de entender, asegurando que todos los involucrados estén alineados con los requisitos del sistema.

Herramientas de Modelado:

- En años recientes, han surgido herramientas para ayudar en la creación de diagramas. Una de ellas es draw.io, un software popular y de código abierto.
- Aunque draw.io es versátil y ampliamente usado para muchos tipos de diagramas, puede que no esté específicamente diseñado para los requisitos de creación de UITDs.

Propósito del Estudio:

- Surge la pregunta de cuán eficiente es crear UITDs usando draw.io en comparación con una herramienta especializada como el editor UITD.
- Para responder a esto, se realizó un estudio donde los participantes crearon UITDs utilizando ambas herramientas, con el objetivo de comparar su eficiencia.

En resumen, la introducción establece el contexto y la relevancia del Diagrama de Transición de Interfaz de Usuario (UITD) como una herramienta esencial en el modelado de interacciones usuario-sistema. Posteriormente, plantea la pregunta central del artículo sobre qué herramienta es más efectiva para crear estos diagramas, lo cual motiva el estudio comparativo presentado en el artículo.

GN

Extrae las palabras principales



Las palabras y frases principales de la introducción son:



1. **Diagrama de Transición de Interfaz de Usuario (UITD)**
2. **Notación de modelado formal**
3. **Simplificar la especificación y diseño**
4. **Interacciones entre el usuario y el sistema**
5. **Flujo de las interfaces de usuario**
6. **Herramienta de comunicación**
7. **Partes interesadas** (técnicas y no técnicas)
8. **Fase de obtención de requisitos**
9. **Precisión técnica**
10. **Desarrollo del sistema**
11. **Simplicidad**
12. **Evidencia empírica**
13. **Herramientas de modelado**
14. **draw.io** (software de diagramación)
15. **Código abierto**
16. **Eficiencia**
17. **Estudio comparativo**

Estas palabras y frases resumen los puntos esenciales y conceptos clave presentados en la introducción del artículo.



Para transformar el texto del artículo al formato Markdown, debemos usar la estructura básica de este lenguaje. Aquí te proporciono el texto en formato Markdown, basándome en los fragmentos que nos has compartido anteriormente:



markdown

Copy code

```
# Saving Time with the User Interface Transition Diagrams (UITD) Editor

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

_Department of Applied Mathematics and Systems_
_Universidad Autonoma Metropolitana_
México City, México

**Emails**:
- jcervantes@cua.uam.mx
- dragonnomadal23@gmail.com
- mgomez@cua.uam.mx

## Abstract
The User Interface Transition Diagram (UITD) is a formal modeling notation t

**Keywords**: Editor Efficiency; Modelling notation; Requirements elicitation

## I. INTRODUCTION

The User Interface Transition Diagram (UITD) is a formal modeling notation t

Simplicity is key when working with modeling notations, particularly for non

In recent years, various tools have been developed to support the creation o

To investigate this question, we conducted a study in which participants wer

(Note: ... indicates that the text continues beyond this point.)
```

Este código en Markdown estructura el artículo con títulos, autores, secciones y formatos correspondientes. Por supuesto, esto es solo una representación basada en los fragmentos que nos has dado. Si tienes todo el artículo o más secciones, puedes seguir la misma estructura para convertirlo completamente a Markdown.



En español



iPor supuesto! Aquí tienes la versión en español del artículo en formato Markdown, basada en la información proporcionada:



markdown

Copy code

```
# Ahorro de Tiempo con el Editor de Diagramas de Transición de Interfaz de U
```

```
## Autores
```

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

```
_Departamento de Matemáticas Aplicadas y Sistemas_
```

```
_Universidad Autonoma Metropolitana_
```

```
Ciudad de México, México
```

```
**Correos Electrónicos**:
```

- 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 m
```

```
**Palabras clave**:
```

 Eficiencia del Editor; Notación de Modelado; Obtención d

```
## I. INTRODUCCIÓN
```

```
El Diagrama de Transición de Interfaz de Usuario (UITD) es una notación de m
```

```
La simplicidad es clave al trabajar con notaciones de modelado, particula
```

```
En años recientes, se han desarrollado diversas herramientas para apoyar la
```

```
Para investigar esta pregunta, realizamos un estudio en el cual se pidió a l
```

```
(Nota: ... indica que el texto continúa más allá de este punto.)
```

Esta versión en español sigue la misma estructura que la versión en inglés, pero con el texto traducido y adaptado al contexto hispanohablante.



Convertir el texto a formato LaTeX es un proceso un poco más detallado, debido a la estructura y el marcado que utiliza LaTeX. Aquí te proporciono una versión básica del artículo en LaTeX basada en los fragmentos que nos has compartido:



latex

Copy code

```
\documentclass[12pt,a4paper]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[spanish]{babel}
\usepackage{graphicx}

\title{Ahorro de Tiempo con el Editor de Diagramas de Transición de Interfaz}
\author{Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C. \\
Departamento de Matemáticas Aplicadas y Sistemas \\
Universidad Autonoma Metropolitana \\
Ciudad de México, México}
\date{}

\begin{document}

\maketitle

\begin{abstract}
El Diagrama de Transición de Interfaz de Usuario (UITD) es una notación de m
\end{abstract}

\textbf{Palabras clave}: Eficiencia del Editor; Notación de Modelado; Obtenc

\section{Introducción}

El Diagrama de Transición de Interfaz de Usuario (UITD) es una notación de m

La simplicidad es clave al trabajar con notaciones de modelado. El UITD prop

En años recientes, se han desarrollado diversas herramientas para apoyar la

\section{Método}

Para investigar esta cuestión, realizamos un estudio en el cual se pidió a 1

% Aquí puedes continuar con las otras secciones del artículo

\end{document}
```

Este es un esquema básico de cómo se vería el artículo en LaTeX. Puedes agregar paquetes adicionales según tus necesidades, por ejemplo, para manejar tablas, imágenes, referencias bibliográficas, etc. También, si tienes todo el artículo o más secciones, puedes seguir la misma estructura para convertirlo completamente a LaTeX.



Si deseas generar un archivo PDF a partir de un archivo escrito en Markdown, puedes utilizar herramientas como Pandoc, que es un convertidor de documentos universales que puede traducir entre varios formatos de marcado, incluido Markdown a PDF. Aquí te mostraré cómo hacerlo en Google Colab:



1. Instala Pandoc y LaTeX (necesario para generar PDFs) en tu entorno de Google Colab:

python

Copy code

```
!apt-get install pandoc
!apt-get install texlive-xetex texlive-fonts-recommended texlive-generic-rec
```

1. Guarda el contenido de Markdown en un archivo llamado `articulo.md`:

python

Copy code

```
%%writefile articulo.md
# Ahorro de Tiempo con el Editor de Diagramas de Transición de Interfaz de U

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

_Departamento de Matemáticas Aplicadas y Sistemas_
_Universidad Autonoma Metropolitana_
Ciudad de México, México

**Correos Electrónicos**:
- jcervantes@cua.uam.mx
- dragonnomadal23@gmail.com
- mgomez@cua.uam.mx

## Resumen
El Diagrama de Transición de Interfaz de Usuario (UITD) es...
... Nuestro estudio tuvo como objetivo comparar la eficiencia de las dos her
```

1. Utiliza Pandoc para convertir el archivo Markdown a PDF:

python

Copy code

```
!pandoc -s articulo.md -o articulo.pdf
```


1. Descarga el archivo PDF generado:

python

Copy code

```
from google.colab import files
files.download('articulo.pdf')
```

Al ejecutar estos pasos en Google Colab, se descargará un archivo PDF llamado `articulo.pdf` con el contenido del artículo en Markdown. Es importante que adaptes el contenido del archivo `articulo.md` en el paso 2 con tu contenido completo de Markdown.

✓ 0s  !pandoc -s articulo.md -o articulo.pdf

✓ 0s [7] from google.colab import files
files.download('articulo.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.

Departamento de Matemáticas Aplicadas y Sistemas
Universidad Autonoma Metropolitana
Ciudad de México, México

Correos Electrónicos: - 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 las interacciones usuario-sistema. Es una valiosa herramienta de comunicación para los interesados técnicos y no técnicos durante la fase de obtención de requisitos, ya que proporciona una notación simple y completa fácil de entender. En este artículo, investigamos la eficiencia de crear UITDs usando draw.io, un software de diagramación ampliamente utilizado, en comparación con un editor UITD especializado. 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, destacando 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 otros interesados involucrados en la especificación y diseño de interacciones usuario-sistema.

Palabras clave: 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 diseño de interacciones usuario-sistema. Puede ser utilizado para modelar el flujo de interfaces de usuario que el sistema tendrá, convirtiéndolo en una valiosa herramienta de comunicación para interesados técnicos y no técnicos durante la fase de obtención de requisitos. Además, es técnicamente preciso, lo que permite que se utilice de manera fiable para iniciar el desarrollo del sistema.

La simplicidad es clave al trabajar con notaciones de modelado, particularmente para no expertos. El UITD proporciona una notación que es completa y suficientemente simple para que los interesados no técnicos la entiendan. Es una

notación de modelado formal fácil de aprender, que facilita la comunicación entre los interesados con y sin habilidades técnicas en desarrollo de software. Comparado con otras herramientas de modelado formal, el UITD tiene varias ventajas, incluyendo su simplicidad, completitud y capacidad para modelar interacciones usuario-sistema de manera clara y concisa. También se proporciona evidencia empírica sobre la comprensibilidad del UITD por parte de interesados no técnicos. Al utilizar el UITD, los diseñadores y desarrolladores pueden crear modelos que son técnicamente precisos y fáciles de entender, ayudando a asegurar que todos los interesados estén alineados y satisfechos con la especificación de los requisitos del sistema.

En años recientes, se han desarrollado diversas herramientas para apoyar la creación de diagramas generalizados. Una de ellas es draw.io, un conocido software de diagramación de código abierto. Draw.io es ampliamente utilizado para crear diagramas, flujogramas y otros tipos de representaciones gráficas. Aunque draw.io es una herramienta versátil, puede 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 utilizar una herramienta especializada como el editor UITD.

Para investigar esta pregunta, realizamos un estudio en el cual se pidió a los participantes crear UITDs utilizando tanto draw.io como el editor UITD. Nuestro estudio tuvo como objetivo comparar la eficiencia de las dos herramientas...

(Nota: ... indica que el texto continúa más allá de este punto.)

P102.3 - Caso de Estudio



En google colab, genera el código que extraiga el texto de varias imágenes PNG guardadas localmente usando OCR en Python



Para extraer texto de imágenes usando OCR (Optical Character Recognition) en Python, una de las bibliotecas más populares es `pytesseract`. A continuación, te mostraré cómo hacerlo en Google Colab:

1. Primero, necesitas instalar las herramientas adecuadas. En Google Colab, puedes hacerlo de la siguiente manera:

```
python
!apt install tesseract-ocr
!pip install pytesseract
```

1. Luego, debes cargar tus imágenes PNG a Google Colab. Una forma fácil de hacerlo es usando la función `files.upload()` de la biblioteca `google.colab`:

```
python
from google.colab import files

uploaded_files = files.upload()
```

1. Una vez que hayas cargado tus imágenes, puedes procesar cada imagen con `pytesseract`:

```
python
import pytesseract
from PIL import Image
from io import BytesIO

for file_name, file_bytes in uploaded_files.items():
    image = Image.open(BytesIO(file_bytes))
    extracted_text = pytesseract.image_to_string(image)

    print(f"Texto extraído de {file_name}:\n")
    print(extracted_text)
    print("="*50)
```

¡Y eso es todo! Ahora deberías ver el texto extraído de cada imagen impreso en tu notebook de Google Colab.

Nota: La precisión del OCR depende de la calidad y claridad de las imágenes que estés procesando. Es posible que no funcione perfectamente en todas las imágenes, pero debería darte un buen punto de partida.

```

7 import pytesseract
from PIL import Image
from io import BytesIO

for file_name, file_bytes in uploaded_files.items():
    image = Image.open(BytesIO(file_bytes))
    extracted_text = pytesseract.image_to_string(image)

    print(f"Texto extraído de {file_name}:\n")
    print(extracted_text)
    print("="*50)

```

integrante del mismo. Dichas modificaciones obligaran a los signatarios a partir de la fecha de su firma.

=====

Texto extraído de Imagen7 (1).png:

CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.
1200353

VIGESIMA SEGUNDA. RECONOCIMIENTO CONTRACTUAL. El presente Contrato constituye el acuerdo integro entre las partes en relación con su objeto y deja sin efecto cualquier otra negociación, obligación o comunicación entre éstas, ya sea verbal o escrita, efectuada con anterioridad a la fecha en que se firme el mismo

VIGESIMA TERCERA. PROCEDIMIENTOS PARA RESOLUCION DE CONTROVERSIAS, DISTINTOS AL PROCEDIMIENTO DE CONCILIACION PREVISTO EN ESTA LEY. Las partes convienen que, en el caso de existir controversias técnicas o administrativas con el objeto y cumplimiento del presente contrato, estas seran resueltas conjuntamente por "EL CIMAV" y "EL PRESTADOR" y en caso de no existir acuerdo al respecto, se estara a lo que disponga "EL CIMAV", obligandose "EL PRESTADOR" a subsanar cualquier deficiencia en un plazo no mayor a diez dias naturales siguientes a la fecha en que reciba la indicación por parte de "EL CIMAV".

VIGESIMA CUARTA. NORMATIVIDAD APLICABLE. "EL CIMAV" y "EL PRESTADOR?" se obligan a sujetarse estrictamente para la adquisicion y entrega de los bienes objeto de este contrato, a todas y cada una de las clausulas que lo integran, a la Ley de Adquisiciones, Arrendamientos y Servicios del Sector Publico, su reglamento; al Código Civil Federal; la Ley Federal de Procedimiento Administrativo; al Código Federal de Procedimientos Civiles; a la Ley Federal de Presupuesto y Responsabilidad Hacendaria y su Reglamento, y a las demas disposiciones juridicas aplicables.

VIGESIMA QUINTA JURISDICCION. Los licitantes o proveedores que infrinjan las disposiciones de la Ley aplicable y seran sancionados por la Secretaria de la Función Publica, de acuerdo a los articulos 59, 60, 61, 62, 63 y 64 de esta ley. Para la interpretacion y cumplimiento del presente Contrato, así como para todo lo no previsto en el mismo, las partes se someten a la jurisdicción de los Tribunales Federales con residencia en la ciudad de Chihuahua, Chihuahua, por lo que renuncian al fuero que por razón de su domicilio presente o futuro pudiera corresponderles.

PREVIA LECTURA DEL CONTENIDO DEL PRESENTE CONTRATO, LAS PARTES ACEPTAN QUEDAR OBLIGADAS EN TODOS SUS TERMINOS Y CONDICIONES, LOS CUALES SE CONSTITUYEN EN LA EXPRESION COMPLETA DE SU VOLUNTAD Y LO FIRMAN DE CONFORMIDAD EN TRES TANTOS, EN 7 FOJAS UTILES, EN LA CIUDAD DE CHIHUAHUA, ESTADO DE CHIHUAHUA, EL 02 DE JULIO DE 2020.

MTRA.
"ELE "REPRESENTAI ITE LEGAL

CONTRATO DE PRESTACIÓN DE SERVICIOS A PRECIO FIJO, QUE CELEBRAN POR UNA PARTE, EL CENTRO DE INVESTIGACIÓN EN MATERIALES AVANZADOS S.C., A QUIEN EN LO SUCESIVO SE LE DENOMINARÁ "EL CIMAV", REPRESENTADA EN ESTE ACTO POR LA MTRA. EVA ACEVEDO VILLAFUERTE; Y POR LA OTRA PARTE, EQUIPOS ANALITICOS Y CIENTIFICOS, S.A. DE C.V., A QUIEN SE LE DENOMINARÁ "EL PRESTADOR", REPRESENTADO EN ESTE ACTO POR EL C.MOISÉS GARCÍA VANEGAS, CONFORME A LAS DECLARACIONES Y CLÁUSULAS SIGUIENTES:

DECLARACIONES:

I. DECLARA "EL CIMAV":

1. Ser una empresa de Participación Estatal Mayoritaria con el carácter de Entidad Paraestatal Federal, creado por dictamen de la Secretaría de Hacienda y Crédito Público en su XXVI vigésima sexta sesión ordinaria de fecha 31 de agosto de 1994, que se encuentra constituida en Escritura Pública Número 42, del Protocolo del Patrimonio Inmueble Federal Número dos, otorgada ante la Fe del Lic. Juan José Royo Provencio, en funciones de Notario por separación del Titular de la Notaría Pública Número 12, Lic. Armando Herrera Acosta, en fecha 25 de octubre de 1994, y registrada bajo el número 233, folios 143, del Libro 36 de la Sección Cuarta, del Registro Público de la Propiedad de este Distrito Judicial Morelos, Estado de Chihuahua, con fecha 23 de marzo de 1995.
2. Que su objeto social de acuerdo a su Instrumento Jurídico de Creación es:
 - I. Realizar actividades de investigación básica y aplicada en las áreas de Ciencia de Materiales y Ciencia y Tecnología Ambiental y disciplinas afines, orientadas hacia la solución de problemas nacionales, regionales y locales del país;
 - II. Formular, ejecutar e impartir programas para estudios de licenciatura, especialidad, maestría, doctorado y estancias posdoctorales, así como cursos de actualización y especialización de personal en actividades relacionadas con el objeto de la Sociedad;
 - III. Orientar la investigación científica y el desarrollo e innovación tecnológica a la modernización del sector productivo y promover y gestionar ante las organizaciones públicas, sociales y privadas, la transferencia del conocimiento, en términos de lo que para el efecto se establezca en la normatividad aplicable;
 - IV. Difundir y publicar información técnica y científica sobre los avances que registre en su especialidad, así como de los resultados de las investigaciones y trabajos que realice;
 - V. Contribuir con el Sistema Nacional de Ciencia y Tecnología a que se refiere la Ley de Ciencia y Tecnología en congruencia con el programa sectorial y la red nacional de grupos y centros de investigación, para asociar el trabajo científico y la formación de recursos humanos de alto nivel al desarrollo del conocimiento y a la atención de las necesidades de la sociedad mexicana;
 - VI. Realizar labores de vinculación con los sectores público, social y privado; y
 - VII. Colaborar con las autoridades competentes en actividades de promoción de la metrología, el establecimiento de normas de calidad y la certificación en apego a la Ley de la materia.
3. Que la **Mtra. Eva Acevedo Villafuerte**, cuenta con las facultades conferidas en el Mandato General para Pleitos y Cobranzas y Actos de Administración otorgado por la Directora General del Centro de Investigación en Materiales Avanzados S.C., en fecha 29 de enero de 2020, la cual consta en la Escritura Pública número 23,400 pasada ante la fe del

Licenciado Hiram Quezada Anchondo, en funciones de Notario por ausencia de su titular el Licenciado Sergio Granados Pineda, Notario Público Número Tres del Distrito Judicial Morelos, contando con las facultades conferidas en el artículo 48 de la Ley de Adquisiciones, Arrendamientos, y Servicios del Sector Público.

4. Que es su voluntad celebrar el presente contrato a fin de que **"EL PRESTADOR"** le suministre los bienes o servicios que se describen en la orden de compra número **1A200353**.
5. Que la adjudicación del presente contrato se realizó de acuerdo al artículo **42 de la Ley de Adquisiciones, Arrendamientos y Servicios Públicos**, y para cubrir las erogaciones que se deriven de este Contrato, se afectará la partida presupuestal número **35701** del proyecto **70012**.
6. Que su Registro Federal de Contribuyentes es **CIM941025MJ1**.
7. Que para la prestación de los servicios objeto del presente Contrato, señala como domicilio el ubicado en **Calle Miguel de Cervantes número 120 del Complejo Industrial Chihuahua en la ciudad de Chihuahua, Chihuahua**.

II. DECLARA "EL PRESTADOR":

1. Que es una **Sociedad anónima de Capital Variable** debidamente constituida conforme a las leyes de los Estados Unidos Mexicanos, en términos de la Escritura Pública número **78,248** de fecha **19 de febrero** del año **2007**, otorgada ante la fe del Licenciado **Jorge Alfredo Domínguez Martínez**, Notario Público número **140** de la Ciudad de México, cuyo primer testimonio se encuentra debidamente inscrito en el Registro Público de la Propiedad y del Comercio bajo el folio mercantil que le correspondió.
2. Que el **C. Moisés García Vanegas** cuenta con las facultades necesarias y suficientes para actuar con la representación legal de la mencionada empresa, las cuales le fueron otorgadas en su carácter de Representante y Apoderado Legal, misma que bajo protesta de decir verdad manifiesta que no le han sido revocadas, modificadas o condicionadas en forma alguna, por lo que cuenta con la personalidad y capacidad legal.
3. Que tiene capacidad jurídica para obligarse de conformidad con el objeto de este instrumento según se desprende de la Escritura Pública número **78,248** de fecha **19 de febrero** del año **2007**, otorgada ante la fe del Licenciado **Jorge Alfredo Domínguez Martínez** Notario Público número **140** del ahora Ciudad de México;
4. Que dispone de la organización, experiencia y demás elementos técnicos, económicos y humanos suficientes para ello, y que conoce en su integridad el material de este contrato, para llevar a efecto los servicios.
5. Que cuenta con los registros, licencias y permisos necesarios para actuar como Prestador del Servicio que origina su contratación a través del presente documento
6. Bajo protesta de decir verdad, manifiesta que no se encuentra en ninguno de los supuestos establecidos en los artículos 29 fracción VIII, 50 y 60 penúltimo párrafo de la Ley de Adquisiciones, Arrendamientos y Servicios del Sector Público.
7. Que es su deseo celebrar el presente contrato y bajo protesta de decir verdad, manifiesta que se encuentra al corriente en el cumplimiento de sus obligaciones fiscales, en términos de lo dispuesto por el artículo 32-D del Código Fiscal de la Federación.

8. Que para los efectos del presente Contrato señala como su domicilio legal el ubicado en la Calle Tenayuca número 287-102 Santa Cruz Atoyac Delegación Benito Juárez, Ciudad de México.
9. Que su Registro Federal de Contribuyentes es E4C0702192S4.

III. DECLARAN AMBAS PARTES:

ÚNICO. Que, vistas las anteriores declaraciones, reconocen mutuamente la personalidad que ostentan y acreditan estar de acuerdo con obligarse, conviniendo en celebrar el presente Contrato al tenor de las siguientes:

CLÁUSULAS

PRIMERA. OBJETO. "EL PRESTADOR" se obliga a proporcionar los servicios de **MANTENIMIENTO Y CONSERVACION DE EQUIPO, MANTENIMIENTO PREVENTIVO AL ESPECROFOTOMETRO DE ABSORCION ATOMICA MARCA GBC, MODELO AVANTAR SIGMA.**

Lo anterior conforme a las características, especificaciones, términos y condiciones que se establecen en el clausulado del presente Contrato, así como a la orden de compra número **1A200353** y la cotización número **200611** emitida para tales efectos por "EL PRESTADOR", remitiéndose en cada momento al contenido de dichos documentos como si se encontrasen insertos en el presente instrumento.

SEGUNDA. PRECIO. El costo de los servicios objeto de este Contrato es de la cantidad de **\$27,700.00 (VEINTISIETE MIL SETESCIENTOS PESOS 00/100 M.N.)** más el 16% correspondiente al Impuesto al Valor Agregado por la cantidad de **\$4,432.00 (CUATRO MIL CUATROSCIENTOS TREINTA Y DOS PESOS 00/100 M.N.)**, dando un total de **\$32,132.00 (TREINTA Y DOS MIL CIENTO TREINTA Y DOS PESOS 00/100 M.N.)**. La cantidad señalada compensará a "EL PRESTADOR" por los materiales, insumos, sueldos, honorarios, organización, dirección técnica y administrativa; prestaciones sociales y laborales a su personal, así como por todos los gastos que se originen como consecuencia de este Contrato, por lo que no podrá exigir mayor retribución por ningún otro concepto.

TERCERA. LUGAR Y FORMA DE PAGO. El monto a que se refiere la cláusula anterior será cubierto por "EL CIMAV" en los términos que este determine dentro de la orden de compra que para tal efecto se eroga. Para ello "EL PRESTADOR" deberá presentar previamente para contra recibo el Comprobante Fiscal Digital (CFDI) en formato PDF y XML correspondiente de acuerdo a las instrucciones de "EL CIMAV", lo cual será ante el **Departamento de Adquisiciones** y con la debida anticipación a la fecha programada para el pago, a fin de que pueda ser revisada y autorizada. El Comprobante Fiscal Digital (CFDI) deberá cumplir con los requisitos fiscales a entera satisfacción de "EL CIMAV".

Los pagos se efectuarán mediante depósito y/o transferencia a la cuenta bancaria habilitada para tales efectos por "EL PRESTADOR."

CUARTA. OBLIGACIONES DE "EL PRESTADOR". Para el debido cumplimiento del objeto materia de este Contrato, "EL PRESTADOR" se obliga a:

- a) Llevar a cabo los servicios a entera satisfacción de "EL CIMAV", en términos de la cotización número **200611** emitida para tales efectos por "EL PRESTADOR" poniendo en juego su experiencia y capacidad.
- b) Informar a "EL CIMAV" sobre sus labores, cuantas veces le sea requerido.
- c) Acudir a prestar los servicios de conformidad con los calendarios, programas y horarios previamente aprobados.

Texto extraído de Imagen1 (1).png:

CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.
14200363

CONTRATO DE PRESTACION DE SERVICIOS A PRECIO FIJO, QUE CELEBRAN POR UNA PARTE, EL CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C., A QUIEN EN LO SUCESIVO SE LE DENOMINARA "EL CIMAV", REPRESENTADA EN ESTE ACTO POR LA MTRA. EVA ACEVEDO VILLAFUERTE; Y POR LA OTRA PARTE, EQUIPOS ANALITICOS Y CIENTIFICOS, S.A. DE C.V., A QUIEN SE LE DENOMINARA "EL PRESTADOR", REPRESENTADO EN ESTE ACTO POR EL C.MOISES GARCIA VANEGAS, CONFORME A LAS DECLARACIONES Y CLAUSULAS SIGUIENTES:

DECLARACIONES:

rE DECLARA "EL CIMAV":

4. Ser una empresa de Participación Estatal Mayoritaria con el caracter de Entidad Paraestatal

Federal, creado por dictamen de la Secretaria de Hacienda y Crédito Publico en su XXVI vigésima sexta sesión ordinaria de fecha 31 de agosto de 1994, que se encuentra constituida en Escritura Publica Numero 42, del Protocolo del Patrimonio Inmueble Federal Numero dos, otorgada ante la Fe del Lic. Juan José Royo Provencio, en funciones de Notario por separación del Titular de la Notaria Publica Numero 12, Lic. Armando Herrera Acosta, en fecha 25 de octubre de 1994, y registrada bajo el numero 233, folios 143, del Libro 36 de la Sección Cuarta, del Registro Publico de la Propiedad de este Distrito Judicial Morelos, Estado de Chihuahua, con fecha 23 de marzo de 1995.

2. Que su objeto social de acuerdo a su Instrumento Juridico de Creación es:

I. Realizar actividades de investigación básica y aplicada en las áreas de Ciencia de Materiales y Ciencia y Tecnología Ambiental y disciplinas afines, orientadas hacia la solución de problemas nacionales, regionales y locales del país;

yf /

II, Formular, ejecutar e impartir programas para estudios de licenciatura, especialidad, maestría, doctorado y estancias posdoctorales, así como cursos de actualización y especialización de personal en actividades relacionadas con el objeto de la Sociedad:

III. Orientar la investigación científica y el desarrollo e innovación tecnológica a la modernización del sector productivo y promover y gestionar ante las organizaciones públicas, sociales y privadas, la transferencia del conocimiento, en términos de lo que para el efecto se establezca en la normatividad aplicable;

. ay)

IV. Difundir y publicar información técnica y científica sobre los avances que registre en su especialidad, así como de los resultados de las investigaciones y trabajos que realice;

V. Contribuir con el Sistema Nacional de Ciencia y Tecnología a que se refiere la Ley de Ciencia y Tecnología en congruencia con el programa sectorial y la red nacional de grupos

y centros de investigación, para asociar el trabajo científico y la formación de recursos humanos de alto nivel al desarrollo del conocimiento y a la atención de las necesidades de la sociedad mexicana;

—
“
aN

VI. Realizar labores de vinculación con los sectores público, social y privado; y

VII. Colaborar con las autoridades competentes en actividades de promoción de la metrología, el establecimiento de normas de calidad y la certificación en apego a la Ley de la materia.

3. Que la Mtra. Eva Acevedo Villafuerte, cuenta con las facultades conferidas en el Mandato General para Pleitos y Cobranzas y Actos de Administración otorgado por la Directora General del Centro de Investigación en Materiales Avanzados S.C., en fecha 29 de enero de 2020, la cual consta en la Escritura Pública número 23,400 pasada ante la fe del

1

=====
Texto extraído de Imagen2 (1).png:

DECLARA “EL PRESTADOR”:

1.

CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.

CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.
14A200353

Licenciado Hiram Quezada Anchondo, en funciones de Notario por ausencia de su titular el Licenciado Sergio Granados Pineda, Notario Público Número Tres del Distrito Judicial Morelos, contando con las facultades conferidas en el artículo 48 de la Ley de Adquisiciones, Arrendamientos, y Servicios del Sector Público.

Que es su voluntad celebrar el presente contrato a fin de que “EL PRESTADOR” le suministre los bienes o servicios que se describen en la orden de compra número 1A200353.

Que la adjudicación del presente contrato se realice de acuerdo al artículo 42 de la Ley de Adquisiciones, Arrendamientos y Servicios Públicos, y para cubrir las erogaciones que se deriven de este Contrato, se afectará la partida presupuestal número 35701 del proyecto 70012.

Que su Registro Federal de Contribuyentes es CIM941025MJ1.

Que para la prestación de los servicios objeto del presente Contrato, señala como domicilio el ubicado en Calle Miguel de Cervantes número 120 del Complejo Industrial Chihuahua en la ciudad de Chihuahua, Chihuahua.

Que es una Sociedad anónima de Capital Variable debidamente constituida conforme a las leyes de los Estados Unidos Mexicanos, en términos de la Escritura Pública número 78,248 de fecha 19 de febrero del año 2007, otorgada ante la fe del Licenciado Jorge — Alfredo Domínguez Martínez, Notario Público número 140 de la Ciudad de México, { i.

cuyo primer testimonio se encuentra debidamente inscrito en el Registro Público de la Propiedad y del Comercio bajo el folio mercantil que le correspondió. \ Ss
Que el C. Moisés García Vanegas cuenta con las facultades necesarias y suficientes para Ss

actuar con la representación legal de la mencionada empresa, las cuales le fueron otorgadas en su carácter de Representante y Apoderado Legal, misma que bajo protesta de decir verdad manifiesta que no le han sido revocadas, modificadas o condicionadas.

🔄 Regenerate

actuar con la representación legal de la mencionada empresa, las cuales le fueron 2 otorgadas en su carácter de Representante y Apoderado Legal, misma que bajo protesta ~~~ de decir verdad manifiesta que no le han sido revocadas, modificadas O condicionadas en s forma alguna, por lo que cuenta con la personalidad y capacidad legal.

Que tiene capacidad jurídica para obligarse de conformidad con el objeto de este instrumento según se desprende de la Escritura Pública número 78,248 de fecha 19 de febrero del año 2007, otorgada ante la fe del Licenciado Jorge Alfredo Domínguez Martínez Notario Público número 140 del ahora Ciudad de México;

Ne

Que dispone de la organización, experiencia y demás elementos técnicos, económicos y humanos suficientes para ello, y que conoce en su integridad el material de este contrato, para llevar a efecto los servicios. e

Que cuenta con los registros, licencias y permisos necesarios para actuar como Prestador del Servicio que origina su contratación a través del presente documento f { :

Bajo protesta de decir verdad, manifiesta que no se encuentra en ninguno de los supuestos establecidos en los artículos 29 fracción VIII, 50 y 60 penúltimo párrafo de la Ley de Adquisiciones, Arrendamientos y Servicios del Sector Público.

Que es su deseo celebrar el presente contrato y bajo protesta de decir verdad, manifiesta que se encuentra al corriente en el cumplimiento de sus obligaciones fiscales, en términos de lo dispuesto por el artículo 32-D del Código Fiscal de la Federación.

=====

Texto extraído de Imagen3 (1).png:

CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.
14200353

8. Que para los efectos del presente Contrato señala como su domicilio legal el ubicado en la Calle Tenayuca número 287-102 Santa Cruz Atoyac Delegación Benito Juárez, Ciudad de México.

9. Que su Registro Federal de Contribuyentes es E4C0702192S4,
III, DECLARAN AMBAS PARTES:

UNICO. Que, vistas las anteriores declaraciones, reconocen mutuamente la personalidad que ostentan y acreditan estar de acuerdo con obligarse, conviniendo en celebrar el presente Contrato al tenor de las siguientes:

CLAUSULAS

PRIMERA. OBJETO. "EL PRESTADOR" se obliga a proporcionar los servicios de \ MANTENIMIENTO Y CONSERVACION DE EQUIPO, MANTENIMIENTO PREVENTIVO AL ESPECTROFOTOMETRO DE ABSORCION ATOMICA MARCA GBC, MODELO AVANTAR SIGMA.

Lo anterior conforme a las características, especificaciones, términos y condiciones que se establecen en el clausulado del presente Contrato, así como a la orden de compra número 1A200353 y la cotización número 200611 emitida para tales efectos por "EL PRESTADOR", remitiéndose en cada momento al contenido de dichos documentos como si se encontrasen insertos en el presente instrumento.

SEGUNDA. PRECIO. El costo de los servicios objeto de este Contrato es de la cantidad de / _ — \$27,700.00 (VEINTISIETE MIL SETESCIENTOS PESOS 00/100 M.N.) más el 16% correspondiente \ al Impuesto al Valor Agregado por la cantidad de \$4,432.00 (CUATRO MIL CUATROSCIENTOS TREINTA Y DOS PESOS 00/100 M.N.), dando un total de \$32,132.00 (TREINTA Y DOS MIL CIENTO TREINTA Y DOS PESOS 00/100 M.N.). La cantidad señalada compensará a "EL PRESTADOR" por los materiales, insumos, sueldos, honorarios, organización, dirección técnica y — administrativa; prestaciones sociales y laborales a su personal, así como por todos los gastos que —

Ss

SS

se originen como consecuencia de este Contrato, por lo que no podrá exigir mayor retribución por ningún otro concepto.

TERCERA. LUGAR Y FORMA DE PAGO. El monto a que se refiere la cláusula anterior será cubierto

por "EL CIMAV" en los términos que este determine dentro de la orden de compra que para tal efecto

se efo. Para ello "EL PRESTADOR" deberá presentar previamente para contra recibo el Comprobante Fiscal Digital (CFDI) en formato PDF y XML correspondiente de acuerdo a las "instrucciones de "EL CIMAV", lo cual será ante el Departamento de Adquisiciones y con la debida anticipación a la fecha programada para el pago, a fin de que pueda ser revisada y autorizada. El Comprobante Fiscal Digital (CFDI) deberá cumplir con los requisitos fiscales a entera satisfacción de

"EL CIMAV"

Los pagos se efectuarán mediante depósito y/o transferencia a la cuenta bancaria habilitada para tales efectos por "EL PRESTADOR" | | t ONG

CUARTA. OBLIGACIONES DE "EL PRESTADOR". Para el debido cumplimiento del objeto materia de

este Contrato, "EL PRESTADOR" se obliga a:

- Llevar a cabo los servicios a entera satisfacción de "EL CIMAV", en términos de la cotización número 200611 emitida para tales efectos por "EL PRESTADOR" poniendo en juego su experiencia y capacidad
- Informar a "EL CIMAV" sobre sus labores, cuantas veces le sea requerido.
- Acudir a prestar los servicios de conformidad con los calendarios, programas y horarios previamente aprobados.



La idea principal de estos textos es la formalización de un contrato de prestación de servicios entre el Centro de Investigación en Materiales Avanzados S.C. (CIMAV) y Equipos Analíticos y Científicos, S.A. de C.V., quien será denominado "EL PRESTADOR". El contrato establece los términos y condiciones para que "EL PRESTADOR" suministre servicios específicos de mantenimiento y conservación de equipo, en este caso, un espectrofotómetro de absorción atómica de una marca específica.

Análisis:

1. **Partes Involucradas:** El contrato describe claramente las partes involucradas y sus representantes. Se ofrecen detalles legales y formales, como registros y facultades, que legitiman a las partes para celebrar este acuerdo.
2. **Objeto del Contrato:** La cláusula primera define claramente el objeto del contrato, que es proporcionar servicios de mantenimiento y conservación de un equipo específico. Se hace referencia a otros documentos, como una orden de compra y una cotización, que forman parte integral del acuerdo.
3. **Precio y Pago:** La cláusula segunda y tercera detalla el precio total por los servicios, incluyendo el Impuesto al Valor Agregado (IVA), y establece las condiciones de pago, como el método de pago y las obligaciones fiscales.
4. **Obligaciones de "EL PRESTADOR":** La cuarta cláusula describe las responsabilidades y obligaciones que "EL PRESTADOR" debe cumplir, como llevar a cabo los servicios de manera satisfactoria y conforme a los horarios previamente acordados.
5. **Contexto Legal:** El documento se enmarca dentro de la ley y normativas aplicables en México, haciendo referencia a leyes específicas como la Ley de Adquisiciones, Arrendamientos y Servicios del Sector Público.
6. **Aspectos Administrativos y Legales:** El contrato incluye una serie de declaraciones y elementos legales que establecen el marco y los precedentes para el acuerdo, como la información de la constitución legal de las partes, registros fiscales, etc.

En resumen, el texto describe un acuerdo legal formal entre dos entidades para la prestación de servicios de mantenimiento de un equipo específico. Las cláusulas detallan las obligaciones, el precio, y los términos bajo los cuales se llevará a cabo el servicio, en un marco legal y regulatorio mexicano. La redacción y estructura del contrato reflejan la seriedad y formalidad requeridas en acuerdos comerciales y legales de esta naturaleza.



Quiénes son las personas involucradas?



Las personas y entidades involucradas en el contrato que proporcionaste son:

1. **CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.**, referido en el contrato como "EL CIMAV".
 - Representado por: **MTRA. EVA ACEVEDO VILLAFUERTE**.
2. **EQUIPOS ANALITICOS Y CIENTIFICOS, S.A. DE C.V.**, referido en el contrato como "EL PRESTADOR".
 - Representado por: **C. MOISES GARCIA VANEGAS**.

Estos son los principales actores mencionados en el contrato que has proporcionado.



Extrae las palabras principales



Del contrato que proporcionaste anteriormente, podemos identificar las palabras o términos clave relacionadas con el objeto y propósito del contrato, así como las entidades involucradas. Las palabras principales son:

1. **CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.** (también abreviado como "EL CIMAV")
2. **EQUIPOS ANALITICOS Y CIENTIFICOS, S.A. DE C.V.** (también referido como "EL PRESTADOR")
3. **CONTRATO DE PRESTACIÓN DE SERVICIOS PROFESIONALES**
4. **MTRA. EVA ACEVEDO VILLAFUERTE**
5. **C. MOISES GARCIA VANEGAS**
6. **SERVICIO**
7. **LICENCIA**
8. **COBERTURA TERRITORIAL**
9. **FECHA DE VIGENCIA**
10. **OBLIGACIONES**
11. **CONDICIONES**
12. **PAGO**
13. **CLAUSULAS**

Estas palabras y términos son centrales para entender el propósito y las obligaciones asociadas con el contrato. Sin embargo, la importancia y relevancia de cada término pueden variar según el contexto y contenido específico de las cláusulas del contrato.



Explica cláusulas



Entendido. Las cláusulas son partes fundamentales en un contrato. Estas son estipulaciones que definen derechos, obligaciones, términos y condiciones para las partes involucradas. Para poder explicarte las cláusulas de manera sencilla, primero necesitaría que me proporcionaras las cláusulas específicas que quieres que explique.

Sin embargo, puedo darte ejemplos de algunas cláusulas comunes y explicarlas de manera simplificada:

1. **Cláusula de confidencialidad:** Esta cláusula significa que ambas partes acuerdan no compartir cierta información con otros. Es como un secreto entre las dos partes.
2. **Cláusula de terminación:** Esta cláusula establece cómo y cuándo se puede finalizar el contrato. Por ejemplo, si una de las partes no cumple con lo que se prometió, la otra parte podría terminar el contrato.
3. **Cláusula de indemnización:** Si algo sale mal debido a las acciones de una de las partes, esa parte se compromete a cubrir los costos o daños. Es como decir: "Si hago algo mal, pagaré por ello".
4. **Cláusula de fuerza mayor:** Esta cláusula dice que si ocurre algo impredecible y fuera de control (como un desastre natural), ninguna de las partes es responsable.
5. **Cláusula de jurisdicción:** Si hay un desacuerdo, esta cláusula define dónde y cómo se resolverá, como en qué ciudad y bajo qué leyes.

Si tienes cláusulas específicas en mente, por favor, compártelas y con gusto te las explicaré de una manera sencilla.



Explicame las cláusulas del contrato proporcionado



Lo siento, pero no me has proporcionado un contrato para revisar. Por lo tanto, no puedo explicarte las cláusulas de ese contrato en particular.

Si tienes un contrato o ciertas cláusulas específicas en mente, por favor, transcribelas aquí, y con gusto te ayudaré a entenderlas de manera sencilla.



Explica de una forma mas sencilla las cláusulas



¡Claro! Aquí tienes una explicación simplificada de las cláusulas del contrato que proporcionaste:



CLAUSULAS:

1. **OBJETO:** "EL PRESTADOR" (EQUIPOS ANALITICOS Y CIENTIFICOS, S.A. DE C.V.) se compromete a hacer un mantenimiento preventivo a un equipo específico, el espectrofotómetro de absorción atómica de la marca GBC, modelo Avantar Sigma, del "EL CIMAV" (Centro de Investigación en Materiales Avanzados S.C.).
2. **PRECIO:** El servicio costará \$27,700 más el 16% de Impuesto al Valor Agregado (IVA), lo que suma un total de \$32,132. Este precio cubre todos los aspectos del servicio, y "EL PRESTADOR" no puede pedir más dinero por ningún otro motivo.
3. **LUGAR Y FORMA DE PAGO:** "EL CIMAV" pagará según lo determine en la orden de compra. "EL PRESTADOR" debe enviar una factura digital en formatos específicos antes del pago. Esta factura debe cumplir con ciertos requisitos fiscales. El pago se hará mediante transferencia bancaria a una cuenta de "EL PRESTADOR".
4. **OBLIGACIONES DE "EL PRESTADOR":**
 - a) Debe realizar el servicio satisfactoriamente, según una cotización específica.
 - b) Debe informar a "EL CIMAV" sobre sus trabajos cuando se le pida.
 - c) Debe seguir un calendario, programas y horarios que se hayan acordado previamente para hacer el servicio.

En resumen, el contrato se trata de un acuerdo donde "EL PRESTADOR" hará un mantenimiento a un equipo de "EL CIMAV" por un precio acordado de \$32,132 y debe seguir ciertas normas y regulaciones al hacerlo.



Genera el código para que reescriba el contrato en formato de Markdown



A continuación, el contrato reescrito en formato Markdown:



markdown

Copy code

```
# CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.
## 14200363

### CONTRATO DE PRESTACION DE SERVICIOS A PRECIO FIJO

Celebrado por:
- **El Centro de Investigación en Materiales Avanzados S.C.** en adelante "
- **Equipos Analíticos y Científicos, S.A. de C.V.** en adelante "EL PRESTA

---

## DECLARACIONES

### Declara "EL CIMAV":
1. Referente a la fundación y constitución legal de la empresa.
2. **Objeto social:**
  - Realizar actividades de investigación en ciencia de materiales y tecnol
  - Impartir programas educativos y de capacitación.
  - Orientar la investigación hacia la modernización del sector productivo.
  - Publicar información técnica y científica.
  - Colaborar con el Sistema Nacional de Ciencia y Tecnología.
  - Vincularse con sectores público, social y privado.
  - Colaborar en actividades de promoción de metrología y certificación.
3. Referente a las facultades de la **Mtra. Eva Acevedo Villafuerte**.

... [resto de las declaraciones] ...

### Declara "EL PRESTADOR":
1. Detalles de la identificación legal y representación.
2. Datos de la constitución legal de la sociedad.
3. Referente a la capacidad jurídica de "EL PRESTADOR".

... [resto de las declaraciones] ...

---

## CLAUSULAS

### PRIMERA. OBJETO
- "EL PRESTADOR" ofrecerá **MANTENIMIENTO Y CONSERVACION DE EQUIPO, MANTENIM

### SEGUNDA. PRECIO
- Costo total: **$32,132.00**.

### TERCERA. LUGAR Y FORMA DE PAGO
- Detalles sobre cómo "EL CIMAV" hará los pagos.

### CUARTA. OBLIGACIONES DE "EL PRESTADOR"
a) Proveer los servicios satisfactoriamente.
b) Informar a "EL CIMAV" de sus labores.
c) Acudir a prestar los servicios según el calendario aprobado.

... [resto de las cláusulas] ...
```

La estructura de markdown puede ser visualizada de manera adecuada en plataformas que soporten este formato, como GitHub, GitLab, o editores de texto que soporten visualización en markdown.

Regenerate



Genera el código en Python para escribir lo generado en Markdown en un archivo PDF



Para generar un PDF desde un contenido escrito en Markdown, primero es necesario convertir ese contenido de Markdown a otro formato que se pueda representar fácilmente en PDF, como HTML. Una vez que el contenido está en formato HTML, se puede convertir fácilmente a PDF.

A continuación, te proporciono un ejemplo de código que realiza esta tarea utilizando las bibliotecas `markdown` para convertir Markdown a HTML y `pdfkit` para convertir HTML a PDF:

python

Copy code

```
import markdown
import pdfkit

# Contenido en Markdown
markdown_content = """
# CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.
## 14200363
...
"""

# Convierte el contenido de Markdown a HTML
html_content = markdown.markdown(markdown_content)

# Convierte el contenido de HTML a PDF
pdfkit.from_string(html_content, 'contrato.pdf')
```

Para ejecutar este código, necesitas instalar las bibliotecas requeridas. Puedes hacerlo con pip:

Copy code

```
pip install markdown pdfkit
```

Además, `pdfkit` utiliza internamente `wkhtmltopdf` para generar el PDF, así que también necesitas instalarlo en tu sistema. Puedes encontrar las instrucciones para tu sistema operativo específico en la [página oficial de wkhtmltopdf](#).

Una vez que hayas instalado las bibliotecas y `wkhtmltopdf`, puedes ejecutar el código arriba y generar el PDF a partir del contenido en Markdown.

Regenerate



Estoy en google colab



¡Entendido! En Google Colab, el proceso es un poco diferente ya que necesitas instalar ``wkhtmltopdf`` en el entorno virtual del colab y hacer algunos ajustes. Aquí está el proceso:



1. Instalar ``wkhtmltopdf`` y las bibliotecas necesarias:

En una celda de Colab, ejecuta:

python

Copy code

```
!apt-get update
!apt-get install -y fonts-liberation libappindicator3-1 libasound2 libatk-br
!wget https://github.com/wkhtmltopdf/packaging/releases/download/0.12.6-1/wk
!apt install ./wkhtmltox_0.12.6-1.bionic_amd64.deb
```

Luego, instala las bibliotecas de Python:

python

Copy code

```
!pip install markdown pdfkit
```

2. Convertir de Markdown a PDF:

En otra celda de Colab, ejecuta:

python

Copy code

```
import markdown
import pdfkit

# Contenido del contrato en Markdown
markdown_content = """
# Contrato de Servicios Profesionales

Este contrato es celebrado entre Cliente y Profesional...

## Cláusula 1: Objeto del Contrato
El profesional se compromete a...

## Cláusula 2: Compensación
El cliente se compromete a pagar...
```

```

## Cláusula 2: Compensación
El cliente se compromete a pagar...

...

"""

# Convertir Markdown a HTML
html_content = markdown.markdown(markdown_content)

# Opciones para el formato del PDF (opcional)
options = {
    'page-size': 'Letter',
    'encoding': "UTF-8",
    'no-outline': None
}


# Indicar la ruta de wkhtmltopdf para pdkit en Colab
path_wkhtmltopdf = '/usr/bin/wkhtmltopdf'
config = pdkit.configuration(wkhtmltopdf=path_wkhtmltopdf)

# Convertir HTML a PDF
pdkit.from_string(html_content, 'contrato.pdf', options=options, configurat

```

Después de ejecutar el código, tendrás un archivo **contrato.pdf** generado en el directorio actual de tu notebook en Colab. Puedes descargarlo directamente desde la interfaz de Colab o utilizando el siguiente código:

python

 Copy code

```

from google.colab import files

files.download('contrato.pdf')

```

Esto debería permitirte convertir tu contrato de Markdown a PDF directamente desde Google Colab.

GN

FileNotFoundError: [Errno 2] No such file or directory: '/usr/bin/wkhtmltopdf'



Mis disculpas por las complicaciones. Parece que hay un problema con la ubicación de ``wkhtmltopdf`` después de la instalación en el entorno de Colab.



Vamos a intentar localizarlo y usar la ruta correcta.

En una celda de Colab, ejecuta:

python

Copy code

```
!which wkhtmltopdf
```

Esto te dará la ruta exacta de ``wkhtmltopdf`` si se ha instalado correctamente. Si la salida es diferente de ``/usr/bin/wkhtmltopdf``, entonces debes usar esa ruta en el código que genera el PDF.

Si, por ejemplo, la salida es ``/usr/local/bin/wkhtmltopdf``, entonces debes actualizar la línea:

python

Copy code

```
path_wkhtmltopdf = '/usr/bin/wkhtmltopdf'
```

a:

python

Copy code

```
path_wkhtmltopdf = '/usr/local/bin/wkhtmltopdf'
```

Luego, intenta nuevamente convertir el Markdown a PDF.

Si no hay salida al ejecutar ``!which wkhtmltopdf``, significa que la instalación de ``wkhtmltopdf`` no fue exitosa. En ese caso, déjame saber y exploraremos otra solución.

GN

!which wkhtmltopdf
no devuelve ninguna ruta



Mis disculpas por el malentendido anterior. Parece que `wkhtmltopdf` no está instalado en el entorno de Google Colab por defecto.

Lo que necesitas hacer es instalarlo manualmente. Sigue estos pasos para hacerlo:

1. Instala `wkhtmltopdf`:

python

Copy code

```
!apt-get update
!apt-get install -y wkhtmltopdf
```

1. Ahora, repite el paso para determinar la ubicación exacta del ejecutable `wkhtmltopdf`:

python

Copy code

```
!which wkhtmltopdf
```

Esto debería devolver una ruta.

1. Ahora usa esa ruta para informar a `pdfkit` dónde buscar el ejecutable y genera tu PDF.

python

Copy code

```
import markdown
import pdfkit

# Markdown content
md_content = """
# Contrato de Servicios

## Partes involucradas:

1. **Proveedor**: Nombre del proveedor.
2. **Cliente**: Nombre del cliente.
```

```

## Detalles del servicio:

- Descripción del servicio.
- Precio: $1000.
- Duración: 30 días.

## Términos y condiciones:

- Término 1.
- Término 2.
- Término 3.
"""

# Convert Markdown to HTML
html_content = markdown.markdown(md_content)

# Define the path for wkhtmltopdf (change this to the output from the previous step)
path_wkhtmltopdf = '/usr/bin/wkhtmltopdf'
config = pdfkit.configuration(wkhtmltopdf=path_wkhtmltopdf)

# Convert HTML to PDF
pdfkit.from_string(html_content, 'output.pdf', configuration=config)

```

Asegúrate de reemplazar ``/usr/bin/wkhtmltopdf`` con la salida que obtuviste en el segundo paso si es diferente.

Esto debería permitirte generar el archivo PDF sin problemas en Google Colab.

 GN

Código en Python para descargar ese PDF




¡Claro! Si quieres descargar el PDF que has generado en Google Colab, puedes usar la biblioteca `files` que viene con `google.colab`.



Aquí tienes un fragmento de código que te permite hacerlo:

python

 Copy code

```
from google.colab import files  
  
files.download('output.pdf')
```

Simplemente ejecuta este código después de haber generado tu PDF y debería iniciarse una descarga del archivo `output.pdf` a tu computadora.

```
[29] !which wkhtmltopdf

/usr/bin/wkhtmltopdf

[35] import markdown
import pdfkit

# Markdown content
md_content = """
# CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.
## 14200963

### CONTRATO DE PRESTACION DE SERVICIOS A PRECIO FIJO

Celebrado por:
- **El Centro de Investigación en Materiales Avanzados S.C.***, en adelante "EL CIMAV", representado por la **Mtra. Eva Acevedo Villafuerte**.
- **Equipos Analíticos y Científicos, S.A. de C.V.***, en adelante "EL PRESTADOR", representado por **C. Moises García Vanegas**.

---

## DECLARACIONES

### Declara "EL CIMAV":
1. Referente a la fundación y constitución legal de la empresa.
2. **Objeto social:**
    - Realizar actividades de investigación en ciencia de materiales y tecnología ambiental.
    - Impartir programas educativos y de capacitación.
    - Orientar la investigación hacia la modernización del sector productivo.
    - Publicar información técnica y científica.
    - Colaborar con el Sistema Nacional de Ciencia y Tecnología.
    - Vincularse con sectores público, social y privado.
    - Colaborar en actividades de promoción de metrología y certificación.
3. Referente a las facultades de la **Mtra. Eva Acevedo Villafuerte**.

... [resto de las declaraciones] ...

### Declara "EL PRESTADOR":
1. Detalles de la identificación legal y representación.
2. Datos de la constitución legal de la sociedad.
3. Referente a la capacidad jurídica de "EL PRESTADOR".

... [resto de las declaraciones] ...

---

## CLAUSULAS

### PRIMERA. OBJETO
- "EL PRESTADOR" ofrecerá **MANTENIMIENTO Y CONSERVACION DE EQUIPO, MANTENIMIENTO PREVENTIVO AL ESPECTROFOTOMETRO DE ABSORCION ATOMICA MARCA GBC, MODELO AVANTAR SIGMA**.

### SEGUNDA. PRECIO
- Costo total: **$32,132.00**.

### TERCERA. LUGAR Y FORMA DE PAGO
- Detalles sobre cómo "EL CIMAV" hará los pagos.

### CUARTA. OBLIGACIONES DE "EL PRESTADOR"
a) Proveer los servicios satisfactoriamente.
b) Informar a "EL CIMAV" de sus labores.
c) Acudir a prestar los servicios según el calendario aprobado.

c) Acudir a prestar los servicios según el calendario aprobado.

... [resto de las cláusulas] ...
"""

# Convert Markdown to HTML
html_content = markdown.markdown(md_content)

# Define the path for wkhtmltopdf (change this to the output from the previous cell)
path_wkhtmltopdf = '/usr/bin/wkhtmltopdf'
config = pdfkit.configuration(wkhtmltopdf=path_wkhtmltopdf)

# Convert HTML to PDF
pdfkit.from_string(html_content, 'Contrato.pdf', configuration=config)

True

[36] from google.colab import files

files.download('Contrato.pdf')
```

CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.

14200363

CONTRATO DE PRESTACION DE SERVICIOS A PRECIO FIJO

Celebrado por: - El Centro de Investigación en Materiales Avanzados S.C., en adelante "EL CIMAV", representado por la Mtra. Eva Acevedo Villafuerte. - Equipos Analíticos y Científicos, S.A. de C.V., en adelante "EL PRESTADOR", representado por C. Moises García Vanegas.

DECLARACIONES

Declara a EL CIMAV:

1. Referente a la fundación y constitución legal de la empresa.
2. **Objeto social:**
3. Realizar actividades de investigación en ciencia de materiales y tecnología ambiental.
4. Impartir programas educativos y de capacitación.
5. Orientar la investigación hacia la modernización del sector productivo.
6. Publicar información técnica y científica.
7. Colaborar con el Sistema Nacional de Ciencia y Tecnología.
8. Vincularse con sectores público, social y privado.
9. Colaborar en actividades de promoción de metrología y certificación.
10. Referente a las facultades de la Mtra. Eva Acevedo Villafuerte.

... [resto de las declaraciones] ...

Declara a EL PRESTADOR:

1. Detalles de la identificación legal y representación.
2. Datos de la constitución legal de la sociedad.
3. Referente a la capacidad jurídica de "EL PRESTADOR".

... [resto de las declaraciones] ...

CLAUSULAS

PRIMERA. OBJETO

- "EL PRESTADOR" ofrecerá **MANTENIMIENTO Y CONSERVACION DE EQUIPO, MANTENIMIENTO PREVENTIVO AL ESPECTROFOTOMETRO DE ABSORCION ATOMICA MARCA GBC, MODELO AVANTAR SIGMA.**

SEGUNDA. PRECIO

- Costo total: **\$32,132.00.**

TERCERA. LUGAR Y FORMA DE PAGO

- Detalles sobre cómo "EL CIMAV" hará los pagos.

CUARTA. OBLIGACIONES DE EL PRESTADOR

a) Proveer los servicios satisfactoriamente. b) Informar a "EL CIMAV" de sus labores. c) Acudir a prestar los servicios según el calendario aprobado.

... [resto de las cláusulas] ...



Y usando pandoc?



¡Usar `pandoc` es una excelente opción! `pandoc` es una herramienta de conversión de documentos universalmente reconocida que puede convertir Markdown a PDF entre muchos otros formatos.



Sigue estos pasos para crear un PDF a partir de Markdown en Google Colab usando `pandoc`:

1. Primero, instala `pandoc` y `texlive` (necesario para la conversión a PDF):

python

Copy code

```
!apt-get install pandoc texlive-xetex
```

1. Luego, crea un archivo Markdown de ejemplo:

python

Copy code

```
contract_md = """
# Contrato

Este es un **contrato** de ejemplo escrito en Markdown.

## Parte 1

Lorem ipsum dolor sit...

## Parte 2

Más texto aquí.
"""

with open("contract.md", "w") as f:
    f.write(contract_md)
```

1. Usa `pandoc` para convertir el archivo Markdown a PDF:

python


Copy code

```
!pandoc -s contract.md -o contract.pdf
```

Regenerate

1. Ahora puedes descargar el archivo usando:

python

 Copy code

```
from google.colab import files  
files.download('contract.pdf')
```

Esto debería proporcionarte un PDF basado en tu texto Markdown usando `pandoc`. Es muy flexible y soporta muchas opciones y estilos, así que si necesitas un formato o diseño específico, `pandoc` probablemente pueda manejarlo con las opciones adecuadas.

CENTRO DE INVESTIGACION EN MATERIALES AVANZADOS S.C.

14200363

CONTRATO DE PRESTACION DE SERVICIOS A PRECIO FIJO

Celebrado por: - El Centro de Investigación en Materiales Avanzados S.C., en adelante “EL CIMAV”, representado por la Mtra. Eva Acevedo Villafuerte. - Equipos Analíticos y Científicos, S.A. de C.V., en adelante “EL PRESTADOR”, representado por C. Moises Garcia Vanegas.

DECLARACIONES

Declara “EL CIMAV”:

1. Referente a la fundación y constitución legal de la empresa.
 2. **Objeto social:**
 - Realizar actividades de investigación en ciencia de materiales y tecnología ambiental.
 - Impartir programas educativos y de capacitación.
 - Orientar la investigación hacia la modernización del sector productivo.
 - Publicar información técnica y científica.
 - Colaborar con el Sistema Nacional de Ciencia y Tecnología.
 - Vincularse con sectores público, social y privado.
 - Colaborar en actividades de promoción de metrología y certificación.
 3. Referente a las facultades de la Mtra. Eva Acevedo Villafuerte.
- ... [resto de las declaraciones] ...

Declara “EL PRESTADOR”:

1. Detalles de la identificación legal y representación.
 2. Datos de la constitución legal de la sociedad.
 3. Referente a la capacidad jurídica de “EL PRESTADOR”.
- ... [resto de las declaraciones] ...

CLAUSULAS

PRIMERA. OBJETO

- “EL PRESTADOR” ofrecerá **MANTENIMIENTO Y CONSERVACION DE EQUIPO, MANTENIMIENTO PREVENTIVO AL ESPECROFOTOMETRO DE ABSORCION ATOMICA MARCA GBC, MODELO AVANTAR SIGMA.**

SEGUNDA. PRECIO

- Costo total: \$32,132.00.

TERCERA. LUGAR Y FORMA DE PAGO

- Detalles sobre cómo “EL CIMAV” hará los pagos.

CUARTA. OBLIGACIONES DE “EL PRESTADOR”

- a) Proveer los servicios satisfactoriamente.
- b) Informar a “EL CIMAV” de sus labores.
- c) Acudir a prestar los servicios según el calendario aprobado.

... [resto de las cláusulas] ...

CONCLUSIONES

Esta práctica respecto a la anterior, me pareció más interesante, porque se logró explorar más a profundidad el uso de ChatGPT con respecto a cosas más técnicas y complicadas, cómo lo es la construcción de código para realizar tareas complicadas como lo son las gráficas y resultados matemáticos, y sobre todo que requieren de librerías en específico para su realización, la versatilidad de ChatGPT permite que pueda ser consultado sobre una amplia gama de tópicos.

Una característica que me resultó especialmente útil fue la capacidad de ChatGPT para adaptarse al entorno donde se trabaja, en este caso, Google Colab. Esta adaptabilidad facilitó una comunicación más fluida, lo que llevó a obtener respuestas más alineadas a las necesidades.

Al principio ChatGPT, por obvias razones no puede saber en qué sistema se está trabajando y aun así te da propuestas, sin embargo, conforme vas usando más la herramienta, aprendes a hacer más específicas las preguntas/tareas, para así evitar resultados que no sirvan para el propósito. Y respecto a la generación de resúmenes, ideas principales, autores, correos, etc... ChatGPT hace muy bien su trabajo al brindar de una forma clara la información y respetando lo que se le pide, sin agregar u omitir detalles.

Sobre mi caso práctico, tuve algunos problemas, ya que llego un momento en que al preguntarle sobre las cláusulas del contrato, me comentó que no le había proporcionado ningún contrato, por lo que tuve nuevamente que mandar el contrato, cuando en otras prácticas sin problema sabía de lo que hablaba aun cuando lo había preguntado mucho antes. También, al momento de pedirle la conversión de Markdown a PDF, me propuso una solución más compleja que la de la práctica P102.2, me salieron algunos errores por las librerías, que al hacer las preguntas necesarias ChatGPT respondió todas ellas claramente y se logró el objetivo. Sin embargo, el PDF generado con ese método no me pareció que tuviera un formato correcto, contrario al de la práctica anterior, dado esto, le hice la pregunta de "¿y usando pandoc?" y me dio la misma solución, que fue más rápida y el PDF fue generado de manera correcta.

La deficiencia que encontré en mi caso fue el largo del texto, no pude enviar el contrato completo, ya que ChatGPT tiene un límite, y esta fue su respuesta en dado caso de que le hubiera enviado el contrato por partes:

Contexto perdido: Al dividir el contrato y enviarlo en partes separadas, podría perder el contexto de las secciones anteriores. Es decir, no "recuerdo" las secciones previas una vez que avanzamos en la conversación. Esto puede afectar la calidad del análisis.

En conclusión, aunque ChatGPT es una herramienta poderosa, el usuario debe ser preciso y claro en sus consultas para obtener resultados óptimos.

