

# ▶ TALENTO TECH



TIC



Universidad  
Tecnológica  
de Bolívar

[www.utb.edu.co/talento-tech](http://www.utb.edu.co/talento-tech)

[www.utb.edu.co/talento-tech](http://www.utb.edu.co/talento-tech)

Proyecto

---

**BOOTCAMP INTELIGENCIA ARTIFICIAL**

**GRUPO: BAS-1038-202406**

**EJECUTOR TÉCNICO:**  
**GUILLERMO BEJARANO REYES**

**MENTOR**

**RODRIGO JUNIOR GARCIA HOYOS**  
**UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR -**  
**UTB**

# Aplicación de Inteligencia Artificial para desarrollar un prototipo de sistema inteligente que utilice técnicas de Machine Learning

## Chatbot Inteligente para la Sugerencia de Precios de Productos y Servicios a Turistas en Cartagena de Indias

### Integrantes

German Ramírez Tamara  
Jesús David Martínez Caamaño  
Jean Lucas Franco Fernández  
Joseph Puello Machado  
Daniel Polo Lidueña

## TABLA DE CONTENIDO

CONTEXTUALIZACIÓN DEL PROBLEMA.....	5
Importancia del Sector. ....	5
Descripción del Problema.....	5
Pregunta de Investigación.....	5
OBJETIVO GENERAL: .....	6
OBJETIVOS ESPECÍFICOS:.....	6
METODOLOGÍA DE TRABAJO .....	7
DEFINIR EL PLANTEAMIENTO DE LA SOLUCIÓN.....	8
Fase 1: Simulación de los precios de productos y servicios. ....	8
Fase 2: Elección de un modelo LLM .....	14
ANÁLISIS DE DATOS.....	17
HERRAMIENTAS A UTILIZAR .....	18
Justificación para su uso. ....	18
Python:.....	18
Librerías usadas: .....	19
Ollama.....	20
CONCLUSIONES DEL MODELO.....	25
REFERENCIAS BIBLIOGRÁFICAS.....	26

## CONTEXTUALIZACIÓN DEL PROBLEMA

Sector para abordar: Turismo

Delimitación territorial: Ciudad de Cartagena de Indias.

### Importancia del Sector.

Cartagena de Indias está ubicada en el norte del Caribe colombiano. Una ciudad tropical mágica encallada en el mar Caribe, vestida de un complejo de fortificaciones de la época colonial que se conservan intactas, esconde una gran riqueza cultural que deviene de múltiples mezclas raciales que le han merecido ser declarada Patrimonio Histórico y Cultural de la Humanidad (UNESCO-1984). Estas características son de gran atractivo para los visitantes nacionales e internacionales, abriendo paso a una industria de gran calado, que representa el 9% del PIB del departamento de Bolívar.

### Descripción del Problema

Recientemente ha sido noticia nacional el abuso al que se ven expuestos los turistas que visitan la ciudad de Cartagena. Este abuso ha tenido casos emblemáticos que han tenido proyección mediática, impactando la imagen reputacional de la ciudad a nivel nacional e internacional, lo que indica falta de políticas públicas de protección al turista y el control de precios y datos en los diferentes sitios de La Heroica.

Este problema ha sido identificado y priorizado por la Alcaldía de Cartagena de Indias de la siguiente manera: *"D20: Acoso al turista y al local por la venta ambulante y estacionaria sin control de precios"* (Plan sectorial de turismo Cartagena de Indias 2024-2027; análisis DOFA)

### Pregunta de Investigación

¿De qué manera un ChatBot impulsado por Inteligencia Artificial Generativa y un modelo de Machine Learning pueden ayudar a los turistas en Cartagena de Indias a identificar precios de referencia y evitar abusos en productos y servicios?

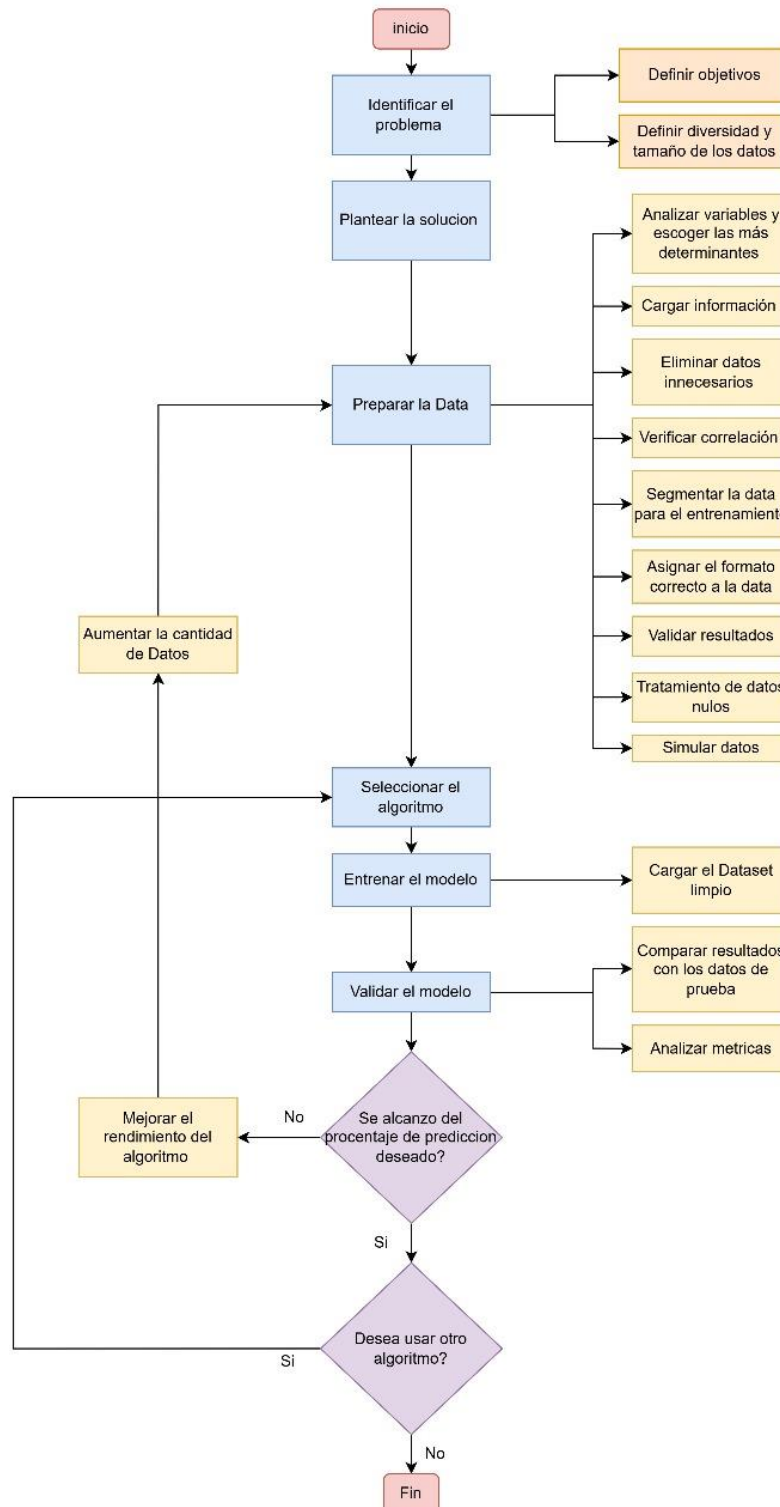
## OBJETIVO GENERAL:

Construir un prototipo de ChatBot impulsado por Inteligencia Artificial Generativa, utilizando un modelo de Machine Learning, que brinde a los turistas de Cartagena de Indias información sugerida y actualizada sobre los precios de productos y servicios, facilitando su experiencia y toma de decisiones.

## OBJETIVOS ESPECÍFICOS:

- **Organizar** los datos históricos simulados de precios de productos y servicios frecuentados por los turistas en Cartagena de Indias para construir un escenario acorde a la realidad.
- **Seleccionar** el modelo de Machine Learning más adecuado para que la IA aprenda los patrones de los precios de referencia.
- **Entrenar** el modelo de Machine Learning utilizando técnicas avanzadas, con el objetivo de mejorar su precisión y capacidad para describir y clasificar precios.
- **Implementar** un sistema interactivo que integre el modelo de Machine Learning con un chatbot, capaz de sugerir precios en tiempo real a los turistas, mejorando su toma de decisiones durante su visita a Cartagena.

## METODOLOGÍA DE TRABAJO





## DEFINIR EL PLANTEAMIENTO DE LA SOLUCIÓN

Se desarrollará una herramienta interactiva basada en un ChatBot impulsado por Inteligencia Artificial Generativa y un modelo de Machine Learning, capaz de proporcionar información en tiempo real sobre los precios de productos y servicios actualizados. El sistema responderá a las consultas de los usuarios, considerando su georeferenciación, con el fin de ofrecer precios sugeridos cercanos a su ubicación y de esta manera ayudar a los turistas a evitar abusos.

De acuerdo al enfoque del proyecto y al no encontrarse una data abierta con datos de empresas y precios de productos/servicios, se procede en realizar 2 fases, las cuales son:

**Fase 1:** Simulación de los precios de productos y servicios.

**Fase 2:** Elección de un modelo LLM (Large Language Model)

### Fase 1: Simulación de los precios de productos y servicios.

Para la Fase 1 se implementa lo siguiente:

- **Clasificación de Inteligencia Artificial:** Generativa
- **Tipo de Inteligencia Artificial:** Descriptiva

#### 1.1.- Preparar la data:

- a. **Recoger datos** de personas naturales y jurídicas inscritas en la Cámara de Comercio de la ciudad de Cartagena de Indias. La fuente proviene del sitio web de Datos Abiertos del Gobierno de Colombia, denominada "BASE DE DATOS DE EMPRESARIOS DE CARTAGENA DESDE EL AÑO 2005" y se encuentra alojada en la siguiente URL:

([https://www.datos.gov.co/Estadisticas-Nacionales/BASE-DE-DATOS-DE-EMPRESARIOS-DE-CARTAGENA-DESDE-EL/5u2a-wah6/about\\_data](https://www.datos.gov.co/Estadisticas-Nacionales/BASE-DE-DATOS-DE-EMPRESARIOS-DE-CARTAGENA-DESDE-EL/5u2a-wah6/about_data)).

- b. **Filtrado de datos:** La Data anterior tiene un total de 22.500 registros y comprende datos desde el año 2005 y su actualización es hasta el 20 de abril de 2024. Sin embargo, en la revisión de los registros se evidenció otras ciudades, por lo cual se realiza filtro de la misma para posteriormente seleccionar las empresas cuyo Código de Actividad Económica (CIIU) estén más relacionadas a los servicios/productos que puede obtener un turista. Los códigos CIIU elegidos para la simulación son:

- 5611: Expendio a la mesa de comidas preparadas
- 5612: Expendio por autoservicio de comidas preparadas



c. **Limpiar la data:** Se procede en revisar los siguientes aspectos:

- **Basura en los datos:** Datos con caracteres no reconocibles.
- **Valores Nulos:** Datos que faltan dentro de cada registro o fila de las fuentes de datos.
- **Valores Duplicados:** Registros o filas de las fuentes de datos que se encuentran repetidas
- **Outliers:** Valores atípicos, muy por encima o muy por debajo del intervalo entre la media y la desviación estándar.

d. **Procesar y Simular la data** de los valores de precios de productos y servicios acorde al punto de referencia; **La Torre del Reloj**, ubicada en el centro histórico de Cartagena, para luego generar una nueva base de datos con precios simulados.

### 1.2.- *Seleccionar el Modelo:*

**Modelo:** Aprendizaje No Supervisado, a través de:

- **Clustering (Técnica General):** Es una técnica de **Machine Learning** no supervisada que agrupa datos en subconjuntos o "**clusters**" según su similitud. El objetivo es que los elementos de un mismo grupo sean lo más parecidos posible, mientras que los de diferentes grupos sean distintos.
- **Método del Codo (Técnica de Evaluación):** Es una técnica para determinar el número óptimo de clusters al usar K-means. El método del codo mide la suma de los errores al cuadrado dentro de los clusters (SSE) para diferentes valores de K.

Modo de ejecución:

- Ejecutar K-means con diferentes valores de K (del 1 a 10).
- Calcular el SSE para cada K.
- Graficas SSE vs. K.

El "**codo**" del gráfico indica el punto donde agregar más clusters no mejora significativamente la agrupación (es decir, se reduce la ganancia marginal).

- **K-means (Algoritmo de clustering):** K-means es un algoritmo de clustering que agrupa los datos en K clusters.

El proceso implica:

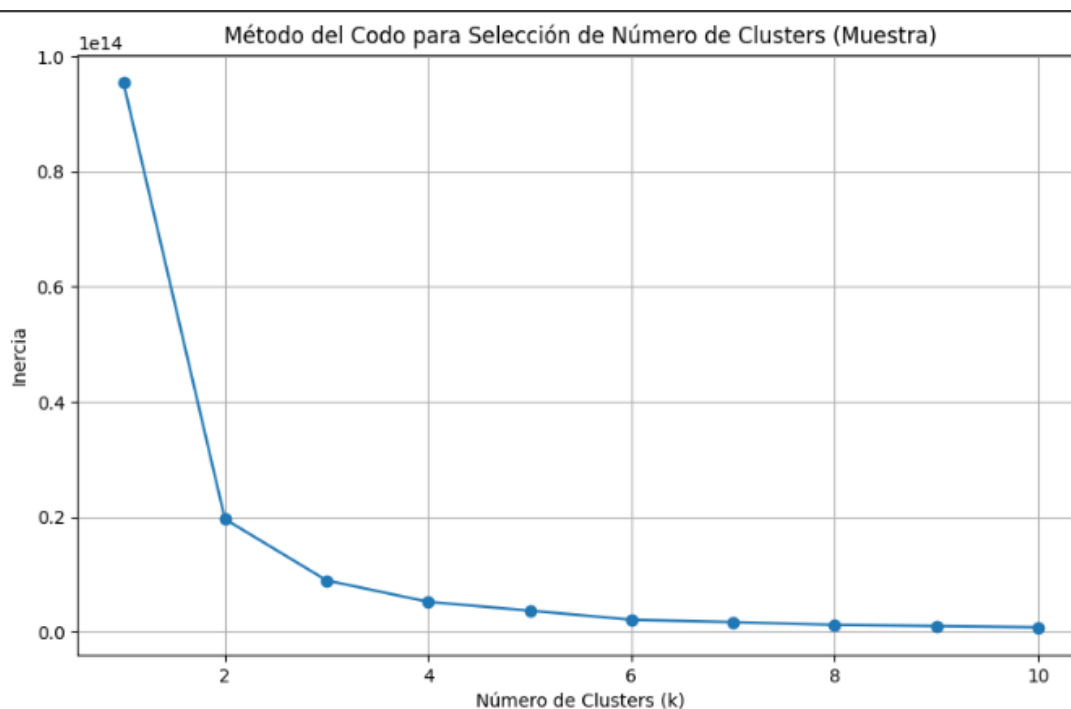
- Elegir K puntos iniciales como centroides.
- Asignar cada dato al clúster cuyo centroide esté más cerca.

- Recalcular los centroides basándose en los datos asignados a cada grupo.
- Repetir los pasos hasta que los centroides no cambien (o se alcance un límite de iteraciones).

### 2.3.- Entrenar el modelo:

De acuerdo a la data simulada, se aprecia en el gráfico del Método del Codo que se puede escoger entre 2,3 o 4 cluster porque son los puntos donde la pendiente comienza a reducirse drásticamente; para lo cual se analizan las 3 opciones en la sección de Probar el Modelo.

Imagen: Método del Codo para Selección de Número de Clusters (Muestra)



Fuente: Propia

### 2.4.- Probar y ajustar el modelo:

Para probar el rendimiento del Algoritmo de Kmeans, se analiza la cantidad de cluster que se pueden usar según el Método del Codo (2,3 o 4) utilizando el Silhouette Score, dando como resultado lo siguiente:

- Con 2 cluster:

```
Coeficiente de Silhouette: 0.76  
El clustering es bueno.
```

- Con 3 cluster

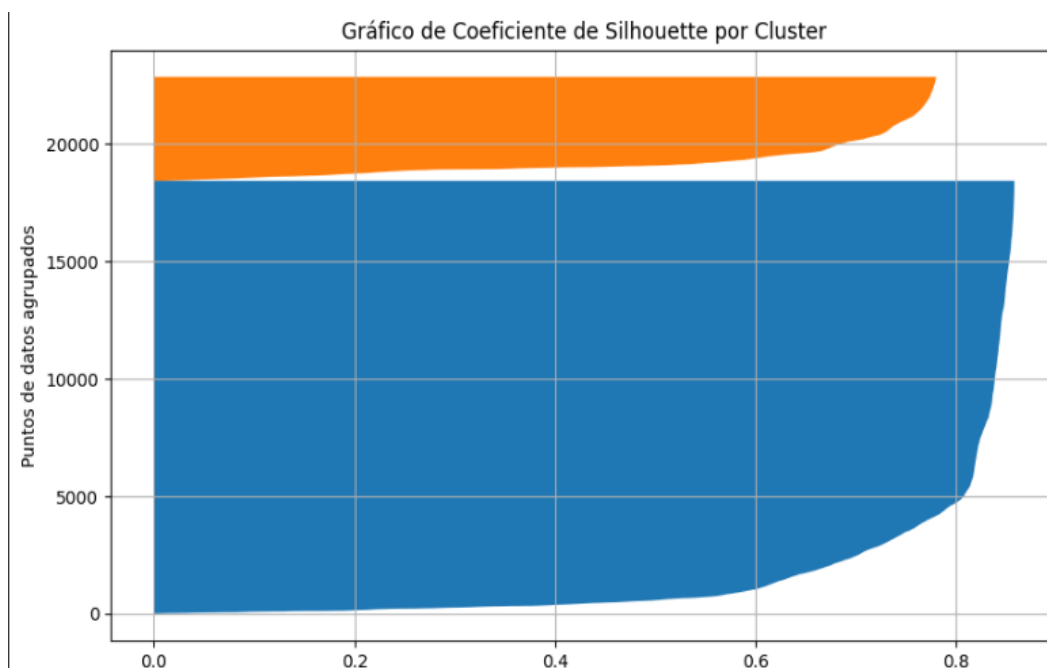
Coeficiente de Silhouette: 0.63  
El clustering es bueno.

- Con 4 cluster

Coeficiente de Silhouette: 0.64  
El clustering es bueno.

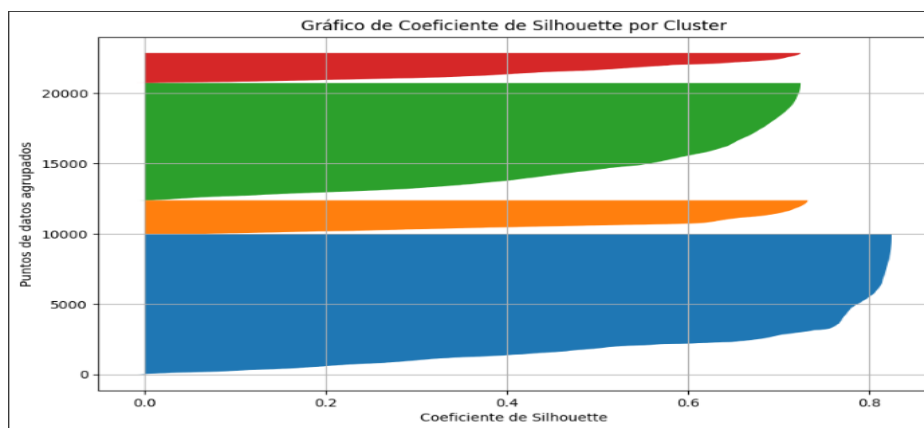
De acuerdo a la prueba realizada, las 3 opciones son buenas, pero la de mayor coeficiente son 2 clusters teóricamente; sin embargo, dado al contexto del proyecto, limitarlo a 2 no capturaría la complejidad de los datos que es una variación entre la Distancia del Punto de Referencia y los Precios de productos/servicios y cómo el objetivo es tener mayor granularidad para analizar patrones más finos sin perder el detrimento de la cohesión interna de los clusters. Se elige la cantidad de 4 cluster, lo cual se puede reflejar en el siguiente análisis de visualización adicional:

Imagen: Gráfico de Coeficiente de Silhouette por 2 Cluster.



Fuente: Propia.

**Imagen: Gráfico de Coeficiente de Silhouette por 4 Cluster.**



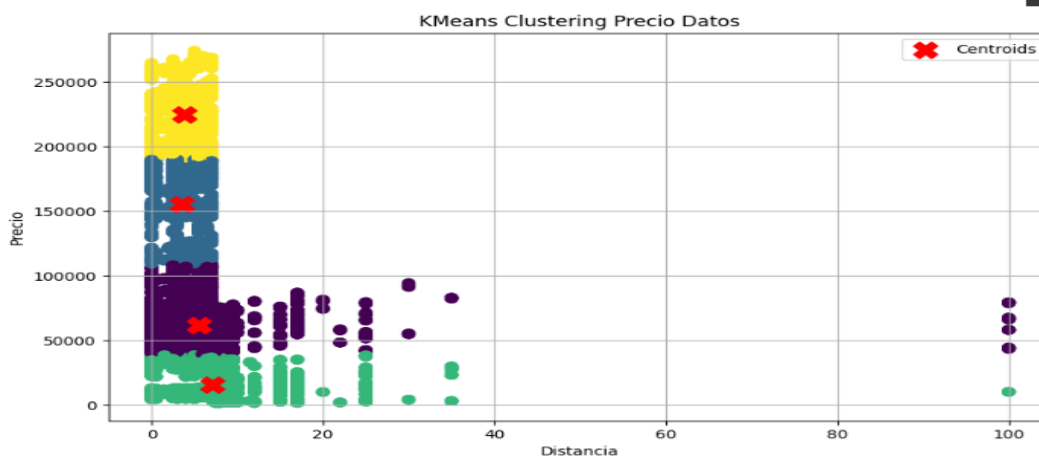
Fuente: Propia

Una vez probado y seleccionada la cantidad de Cluster, se procede a probar y ajustar el Algoritmo K-Means con 4 clusters, del cual se logra contrastar en la gráfica la Distancia (eje x) vs los Precios (eje y) de toda la información simulada.

De igual forma, se puede observar la segmentación de los datos en 4 segmentos de colores distintos y a su vez los centroides identificados con un símbolo (X) de color rojo.

En el análisis de la data, se establece que está acorde a la realidad, toda vez que a mayor cercanía al Punto de Referencia (Torre de Reloj) ubicado en el barrio Centro, el precio de los productos/servicios aumenta. Adicionalmente, que los precios pueden ser superiores en las Islas en comparación con otros barrios de la ciudad.

**Imagen: Algoritmo KMeans Clustering Distancia Vs Precios**



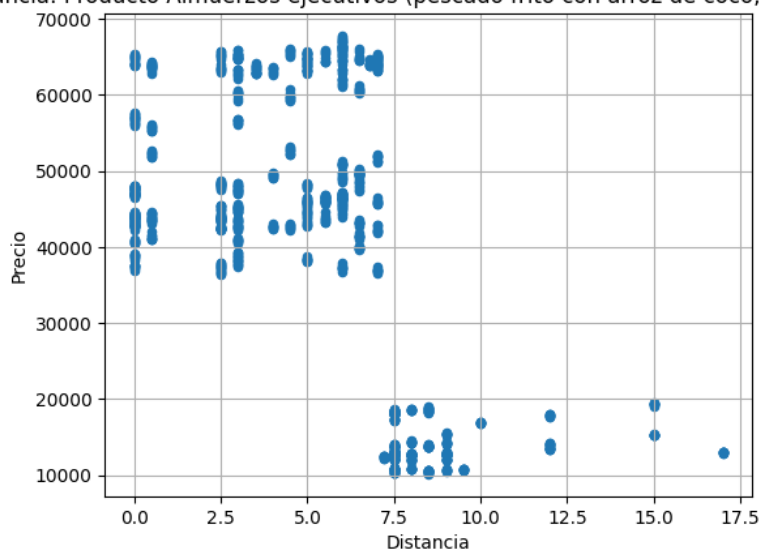
Fuente: Propia

A continuación se realizan pruebas al Algoritmo K-means con los siguientes productos:

Almuerzos ejecutivos (pescado frito con arroz de coco, ensalada y patacones)

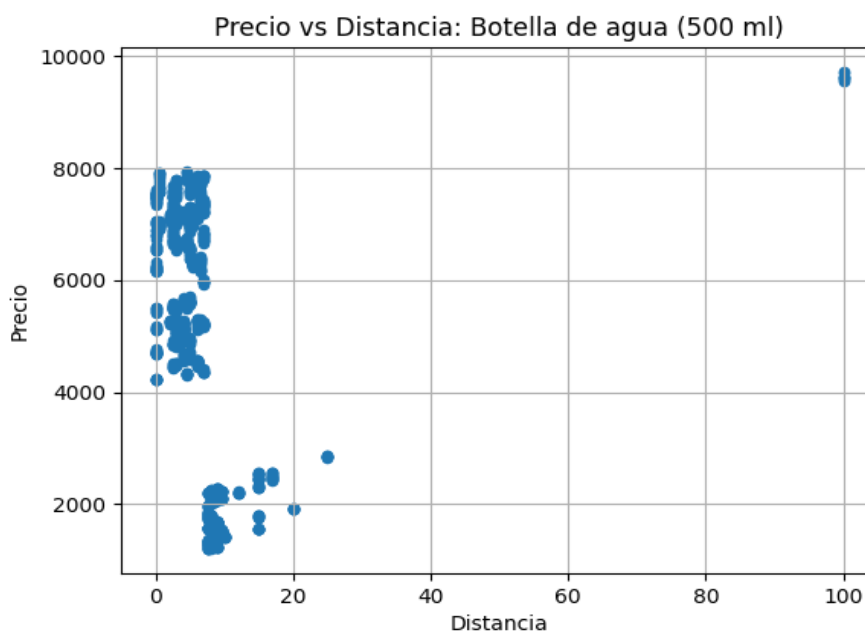
Imagen: Algoritmo KMeans Clustering Distancia Vs Precios

Precio vs Distancia: Producto Almuerzos ejecutivos (pescado frito con arroz de coco, ensalada y patacones)



Fuente: Propia

Imagen: Precio vs Distancia: Botella de agua (500 ml)



Fuente: Propia

## Fase 2: Elección de un modelo LLM

Para la Fase 2 se implementa lo siguiente:

- **Clasificación de Inteligencia Artificial:** Generativa
- **Tipo de Inteligencia Artificial:** Auto-supervisado
- **Estructura del Modelo:** Red Neuronal Profunda
- **Aplicación:** Procesamiento del Lenguaje Natural (NLP)
- **Ejecución:** Local mediante Ollama

### 2.1.- Preparar la data:

Para esta fase, la data ha sido limpiada y depurada en la fase anterior.

### 2.2.- Seleccionar el Modelo:

Para la selección del Modelo, se ha elegido una infraestructura de Ollama; debido a que es Open Source y el servidor se puede implementar de manera local.

Entre los modelos a seleccionar se encuentra el **llama3.2** que incluye Transformer; sin embargo, dado a la capacidad del equipo actual (Procesador Intel Core i5, 8 núcleos a 2.4Ghz de 11th Gen y memoria RAM 16 GB), este no corre, por lo tanto se selecciona un modelo más compacto: **dog/arcee-lite**

### 2.3.- Entrenar el modelo:

Para entrenar el modelo, se diseñó un script en Python en donde se genera un Dataframe y desde el mismo se realizan los filtros de los inputs de Producto y Distancia para que así se pueda calcular precio mínimo, máximo y promedio; finalmente se invoca al Modelo LLM para colocarlo en contexto y de esta manera pueda dar respuesta acorde a la información previa y suministrada por el usuario:

Imagen: Código fuente de la interfaz del Chatbot

```
# Mostrar estadísticas de precios
st.write("Precio Mínimo:", df2.Precio.min())
st.write("Precio Máximo:", df2.Precio.max())
st.write("Precio Promedio:", df2.Precio.mean())
st.dataframe(df2)
#st.write(df2.to_json())
# Configuración del modelo LLaMA
modelo_seleccionado = 'dog/arcee-lite'
context = f'Ponte en rol de un recomendador de productos de la ciudad de cartagena usa solo esta informacion para tus
user_input = st.text_input("Escribe tu pregunta o mensaje:")
```

Fuente: Propia

### 2.4.- Probar y ajustar el modelo:

Para probar el modelo LLM, se realiza pregunta al ChatBot y luego se compara con la información de Dataframe y las variables definidas (precio mínimo, precio máximo y precio promedio); identificando que el Chatbot responde de manera acertada, a continuación, se comparte la prueba realizada:

En la prueba realiza se realiza la consulta al ChatBot sobre el precio que tiene un producto, el cual el responde de acuerdo al contexto de la pregunta ya que el usuario no digito el nombre exacto del producto; la respuesta dada por el ChatBot es el valor mínimo y no el promedio; por lo cual se realiza afinamiento al código.

Imagen: Primera consulta al ChatBot Just Price Bot

Escribe tu pregunta o mensaje:

Que valor tiene un almuerzo ejecutivo

Obtener Respuesta

El valor sugerido del producto Almuerzos ejecutivos (pescado frito con arroz de coco, ensalada y patacones) es de 42027.34 COP.

Fuente: Propia

Imagen: Valores para comprobar la petición

Precio Mínimo: 42027.34

Precio Máximo: 48260.75

Precio Promedio: 44967.433999999994

Fuente: Propia

Imagen: Consulta de lugares recomendados:



Escribe tu pregunta o mensaje:

Cual es el precio de sugerencia de Cremas y lociones after sun

Obtener Respuesta

I

El precio de sugerencia para Cremas y lociones after sun sería:

1. 6736.87 COP (valor de la empresa VERBEL MARTINEZ DAYRON ALONSO)
2. 6842.44 COP (valor de la empresa MAPAMI INVERSIONES S.A.S.)
3. 7772.82 COP (valor de la empresa RUIZ BELTRAN LUIS ANTONIO)
4. 8075.09 COP (valor de la empresa DENG YUCHAN)
5. 8980.66 COP (valor de la empresa GOMEZ RIOS ANTONY)
6. 9287.66 COP (valor de la empresa CARABALLO MARTINEZ MARELVIS TATIANA)
7. 9339.19 COP (valor de la empresa DE ORO MARTINEZ LEONARDO DE JESUS)
8. 9968.04 COP (valor de la empresa PEREZ MARTINEZ TOMASA CECILIA)

Fuente: Propia

De la anterior petición, se evidencia que el ChatBot respondió acorde a la solicitud y listó un total de 8 valores del producto solicitado de manera organizada, es decir desde el menor al mayor; cabe resaltar que las empresas están cercanas a la ubicación de de la persona.

Imagen: Valores para comprobar la petición

## Proyecto de prueba

Distancia a la torre del reloj



1

100

Precios de Productos Seleccionados

Producto	Empresa		Distancia	Precio	Fecha
Cremas y lociones after sun	VERBEL MARTINEZ DAYRON ALONSO		7.5	6,736.87	2024-10-31
Cremas y lociones after sun	MAPAMI INVERSIONES S.A.S.	DTA	8	6,842.44	2024-10-31
Cremas y lociones after sun	RUIZ BELTRAN LUIS ANTONIO	DTA	8	7,772.82	2024-10-31
Cremas y lociones after sun	DENG YUCHAN	RRR	7.5	8,075.09	2024-10-31
Cremas y lociones after sun	GOMEZ RIOS ANTONY		8	8,980.66	2024-10-31
Cremas y lociones after sun	CARABALLO MARTINEZ MARELVIS TATIANA	CAL	7.5	9,287.66	2024-10-31
Cremas y lociones after sun	DE ORO MARTINEZ LEONARDO DE JESUS	AR	8	9,339.19	2024-10-31
Cremas y lociones after sun	PEREZ MARTINEZ TOMASA CECILIA	DTA	8	9,968.04	2024-10-31
Cremas y lociones after sun	GARCIA MOLINA JOSE FERNANDO	NUF	7.5	10,105.64	2024-10-31
Cremas y lociones after sun	GONZALEZ RODRIGUEZ EMIRO DE JESUS	A	7.5	10,425.83	2024-10-31

Fuente: Propia

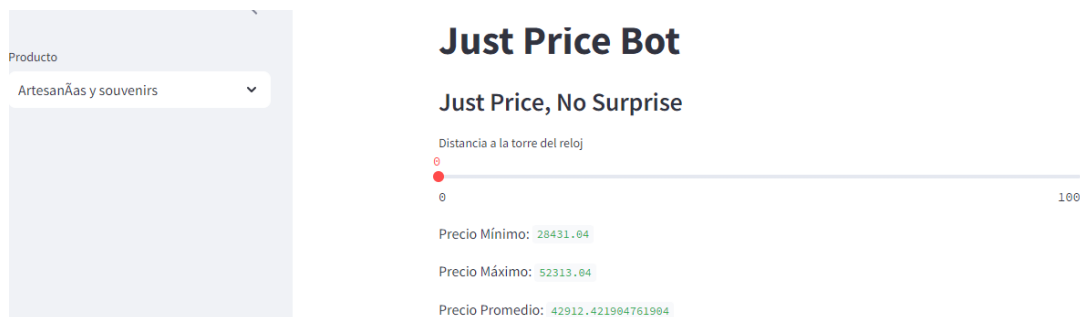
## ANÁLISIS DE DATOS

A continuación, se analiza el comportamiento de valores de precios del mismo producto, pero con diferentes distancias:

**Producto 1, Escenario 1:** Artesanía y Souvenir a 0 Km del Punto de Referencia.  
Datos:

- Precio Mínimo: 28431.04
- Precio Máximo: 52313.04
- Precio Promedio: 42912.421904761904

Imagen: Consulta para Producto 1, Escenario 1

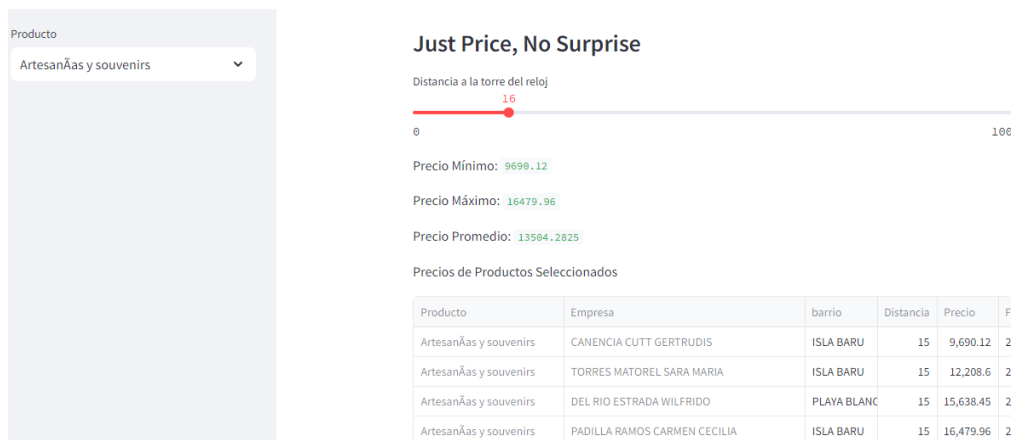


Fuente: Propia

**Producto 1, Escenario 2:** Artesanía y Souvenir a 16 Km del Punto de Referencia.  
Datos:

- Precio Mínimo: 9690.12
- Precio Máximo: 16479.96
- Precio Promedio: 13504.2825

Imagen: Consulta para Producto 1, Escenario 2



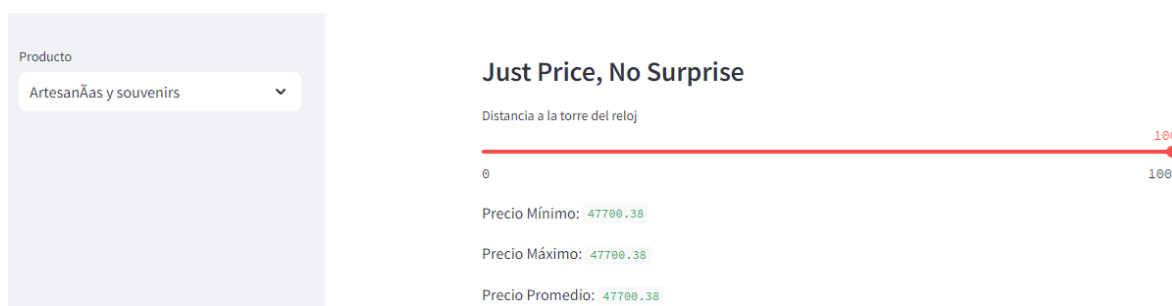
Fuente: Propia

### Producto 1, Escenario 3: Artesanía y Souvenir a 100 Km del Punto de Referencia.

Datos:

- Precio Mínimo: 47700.38
- Precio Máximo: 47700.38
- Precio Promedio: 47700.38

Imagen: Consulta para Producto 1, Escenario 3



Fuente: Propia

De las 3 pruebas anteriores se puede apreciar que entre más cerca se encuentre al punto de referencia, más costosos son los productos y el mismo comportamiento se refleja en las Islas.

## HERRAMIENTAS A UTILIZAR

### Justificación para su uso.

#### Python:

Python es fundamental para el desarrollo de proyectos de machine learning (ML) e inteligencia artificial (IA) por varias razones:

- **Amplia variedad de bibliotecas y frameworks:** Python tiene una gran cantidad de bibliotecas y frameworks especializados en ML e IA, lo que facilita el desarrollo de modelos avanzados.
- **Simplicidad y legibilidad:** Python tiene una sintaxis clara y fácil de entender, lo que hace que los desarrolladores puedan enfocarse más en resolver problemas de ML/IA en lugar de luchar con la complejidad del código. Su simplicidad permite que tanto principiantes como expertos puedan trabajar en proyectos sin muchas barreras.

- **Comunidad activa y soporte:** Python cuenta con una comunidad enorme y activa que continuamente desarrolla nuevas herramientas, soluciones y mejoras. Además, hay una gran cantidad de recursos en línea, como tutoriales, foros y documentación, que ayudan a resolver problemas y aprender conceptos rápidamente.
- **Integración con otras tecnologías:** Python se integra fácilmente con otros lenguajes y tecnologías, lo que permite el desarrollo de sistemas complejos. Se puede combinar con APIs, bases de datos, y otras plataformas para crear aplicaciones completas que utilicen IA.
- **Desarrollo rápido de prototipos:** Gracias a su simplicidad y a la variedad de herramientas disponibles, Python permite construir y probar prototipos de modelos de machine learning rápidamente. Esto es crucial para investigadores y desarrolladores que necesitan iterar y experimentar con distintas soluciones antes de implementar una definitiva.
- **Uso extendido en investigación y ciencia de datos:** Python es la elección principal en el ámbito de la investigación en ciencia de datos, lo que facilita la transición entre la investigación académica y la aplicación industrial. Muchos artículos científicos y proyectos de investigación se publican utilizando Python, y sus herramientas se actualizan constantemente para soportar las últimas innovaciones en IA.
- **Portabilidad y escalabilidad:** Los modelos creados con Python pueden escalar fácilmente para usarse en aplicaciones del mundo real, desde sistemas de recomendación hasta análisis de grandes volúmenes de datos. Además, Python es compatible con diversas plataformas y sistemas operativos.
- **Soporte para big data:** Las herramientas de Python están diseñadas para manejar grandes volúmenes de datos, lo que es esencial para el desarrollo de modelos de machine learning y análisis de datos. Bibliotecas como Dask y PySpark ayudan a procesar y analizar datos a gran escala.

#### Librerías usadas:

- **Random:** Esta librería estándar de Python se utiliza para generar números aleatorios y realizar operaciones relacionadas, como elegir elementos aleatoriamente, mezclar secuencias, entre otras. Es muy útil en simulaciones y pruebas aleatorias.
- **Pandas (pd):** Es una de las bibliotecas más populares para el manejo y análisis de datos en Python. Proporciona estructuras de datos flexibles, como

DataFrames, que permiten manipular, limpiar y analizar datos de manera eficiente.

- **Datetime y timedelta:** Estas clases son parte del módulo datetime de Python y se utilizan para manejar fechas y horas. datetime permite trabajar con fechas y horas específicas, mientras que timedelta se utiliza para realizar operaciones aritméticas con fechas, como sumar o restar días.
- **Calendar:** Este módulo proporciona funciones relacionadas con los calendarios, como obtener el nombre del mes, el día de la semana, e incluso la creación de calendarios completos. Es útil cuando se necesita información detallada sobre las fechas.
- **Math:** La librería math es parte del estándar de Python y proporciona funciones matemáticas avanzadas, como operaciones trigonométricas, exponenciales, logarítmicas, y otros cálculos matemáticos complejos.
- **Sklearn.cluster.KMeans:** Esta función es parte de la biblioteca scikit-learn, una de las más usadas en el campo del machine learning. El algoritmo KMeans se utiliza para el análisis de clustering, agrupando datos en diferentes clusters o grupos según sus características.
- **Matplotlib.pyplot (plt):** matplotlib es una biblioteca popular para la visualización de datos en Python. Su submódulo pyplot proporciona una interfaz simple para crear gráficos y visualizaciones como gráficos de líneas, barras, dispersión y más.
- **Warnings:** Este módulo se utiliza para manejar advertencias en Python, permitiendo mostrar, ocultar o controlar mensajes de advertencia generados durante la ejecución del programa.
- **Streamlit (st):** Streamlit es un marco de trabajo para crear aplicaciones web interactivas y fáciles de usar, especialmente orientadas a la visualización de datos y el machine learning. Permite convertir scripts de Python en aplicaciones web sin necesidad de conocimientos avanzados de desarrollo web.

## Ollama

- **Ejecución local:** Una de las principales ventajas de Ollama es que permite ejecutar modelos de lenguaje directamente en tu máquina, sin necesidad de conectarse a servidores externos. Esto es beneficioso para proyectos que requieren una mayor privacidad o para aquellos que se quieren desarrollar sin depender de una conexión a Internet.

- **Control y personalización:** Al ejecutar el chatbot localmente con Ollama, tienes un control completo sobre el modelo y los datos con los que trabaja. Puedes personalizar el modelo para que se ajuste mejor a las necesidades específicas del proyecto y adaptarlo para que aprenda directamente de la información suministrada.
- **Privacidad de datos:** La privacidad es una preocupación importante cuando se trabaja con datos sensibles. Dado que Ollama permite la ejecución local, no es necesario enviar los datos a servicios en la nube, lo que reduce el riesgo de filtraciones o accesos no autorizados.
- **Compatibilidad con modelos personalizados:** Ollama soporta la integración de modelos de lenguaje que pueden ser entrenados con datos específicos para mejorar su precisión en tareas particulares. Esto es esencial para un chatbot que debe aprender y responder de manera efectiva con la información suministrada por el usuario.
- **Optimización y eficiencia:** Los modelos de lenguaje que se ejecutan localmente con Ollama están optimizados para utilizar los recursos del hardware de tu máquina, lo que puede resultar en un rendimiento más rápido y eficiente en comparación con el uso de APIs de servicios externos.
- **Facilidad de uso:** Ollama está diseñado para ser fácil de implementar y configurar, lo cual es ideal para desarrolladores que quieran crear un chatbot sin tener que lidiar con la complejidad de la implementación de grandes modelos de lenguaje desde cero. Además, la documentación y comunidad de usuarios proporcionan un buen soporte para resolver problemas o mejorar el proyecto.
- **Escalabilidad y flexibilidad:** Aunque tu proyecto comienza local, siempre tienes la opción de escalarlo a entornos más grandes o moverlo a la nube si fuera necesario en el futuro. Ollama ofrece la flexibilidad para que puedas hacer esta transición sin tener que modificar mucho tu código base.
- **Interacción natural y aprendizaje automático:** Ollama puede utilizar modelos de lenguaje avanzados que son capaces de entender y procesar lenguaje natural de manera efectiva. Esto permite que el chatbot proporcione respuestas más naturales y coherentes, mejorando la experiencia del usuario a medida que aprende de los datos proporcionados.



# Desarrollo del proyecto

## 1. Implementación del código en Google Colab.

### Fase 1: Simulación de Precios

```
import random
import pandas as pd
from datetime import datetime, timedelta
import calendar
import math
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")
df = pd.read_csv('./bd_precios_gsheet_08102024.csv')
productos_servicios = df[['producto_servicio', 'precio_unitario']]
productos_servicios.to_csv('productos_servicio.csv')
empresas = df[['razon_social', 'tipo', 'nit_cc', 'dirección', 'barrio']]
empresas.to_csv('empresa.csv')
df_companies = pd.read_csv('empresa.csv')
def calcular_precio_ajustado(base_price, distance):
    if distance <= 7:
        factor = 2 + (distance / 50) * 0.5
    elif distance <= 60:
        factor = 1 + ((distance - 50) / 50) * 0.5
    else:
        factor = 2.5 + math.log(distance - 100 + 1, 3) * 0.5
    return round(base_price * factor, 2)
def generate_monthly_dates(end_date, num_months):
    dates = []
    for i in range(num_months):
        date = end_date - timedelta(days=30*i)
        last_day = calendar.monthrange(date.year, date.month)[1]
        return sorted(dates)
def apply_gradual_price_change(base_price, num_changes):
    prices = [base_price]
    for _ in range(num_changes - 1):
        change_percent = random.uniform(-0.010, 0.010)
        new_price = prices[-1] * (1 + change_percent)
        prices.append(round(new_price, 2))
    return prices
end_date = datetime.now().replace(hour=0, minute=0, second=0,
microsecond=0)
num_months = 12
data = []
for _, company in df_companies.iterrows():
    empresa = company['razon_social']
    distancia = company['distancia_km_del_reloj']
    barrio = company['barrio']
    productos_elegidos = df_products.sample(n=random.choice([3, 5]))
    for _, producto in productos_elegidos.iterrows():
```



```

        precio_base = producto['precio_unitario']
        dates = generate_monthly_dates(end_date, num_months)
        precios_historicos = apply_gradual_price_change(precio_base,
num_months)
        for date, precio_historico in zip(dates, precios_historicos):
            precio_ajustado =
calcular_precio_ajustado(precio_historico, distancia)
            data.append([empresa,barrio, distancia,
producto['producto_servicio'], precio_ajustado, date.strftime("%Y-%m-
%d")])
df_final = pd.DataFrame(data, columns=['Empresa','barrio',
'Distancia', 'Producto', 'Precio', 'Fecha'])
df_final = df_final.sort_values(['Empresa', 'Producto', 'Fecha'])
df_final.to_excel('precios.xlsx', index=False)
X = df_final[['Distancia', 'Precio']]
inertia_sampled = []
K_range = range(1, 11)
for k in K_range:
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(X)
    inertia_sampled.append(kmeans.inertia_)
plt.figure(figsize=(10, 6))
plt.plot(K_range, inertia_sampled, marker='o')
plt.title('Método del Codo para Selección de Número de Clusters
(Muestra)')
plt.xlabel('Número de Clusters (k)')
plt.ylabel('Inercia')
plt.grid(True)
plt.show()
kmeans = KMeans(n_clusters=4, random_state=42)
kmeans.fit(X)
X['Cluster'] = kmeans.labels_
plt.figure(figsize=(10, 6))
plt.scatter(X['Distancia'], X['Precio'], c=X['Cluster'],
cmap='viridis', s=50)
plt.scatter(kmeans.cluster_centers[:, 0], kmeans.cluster_centers[:,
1], s=200, c='red', label='Centroids', marker='X')
plt.title('KMeans Clustering Precio Datos')
plt.xlabel('Distancia')
plt.ylabel('Precio')
plt.legend()
plt.grid(True)
plt.show()

```

## Fase 2: Simulación de Precios

```

import streamlit as st
import pandas as pd
import ollama

# Cargar el archivo de Excel
df = pd.read_excel('precios.xlsx')

```

```
# Configuración de la interfaz de Streamlit
st.title('Proyecto de prueba')
distancia = st.slider('Distancia a la torre del reloj', 1, 100)
producto = st.sidebar.selectbox('Producto', df.Producto.unique())

# Filtrado y procesamiento de datos
distancia_1 = float(distancia) - 1
df2 = df.query(f'Producto == "{producto}" and Distancia <= {distancia}
and Distancia >= {distancia_1}')
df2 = df2.groupby(['Producto', 'Empresa']).max().sort_values('Precio')

modelo_seleccionado = 'llama2-uncensored'
context = f'Ponte en rol de un recomendador de productos de la ciudad
de cartagena usa solo esta informacion para tus respuestas, Precio
Mínimo:{df2.Precio.min()}, Precio Máximo:{df2.Precio.max()}, Precio
Promedio:{df2.Precio.mean()} tabla de negocios{df2.to_json()} estoy a
{distancia} kilometros de la torre del reloj. Responde en español. '
user_input = st.text_input("Escribe tu pregunta o mensaje:")

# Botón para activar la respuesta del modelo LLaMA
if st.button("Obtener Respuesta"):
    if user_input:
        try:
            # Reemplazar con la forma correcta de generar texto usando
            Ollama
            ollama_response =
ollama.generate(model=modelo_seleccionado, prompt=context+'
'+user_input)
            # Mostrar la respuesta del modelo LLaMA
            st.write("Respuesta del Modelo LLaMA:")
            st.write(ollama_response["response"])
        except Exception as e:
            st.write("Ocurrió un error:", e)
    else:
        st.write("Por favor, introduce un mensaje.")
```

### CONCLUSIONES DEL MODELO

La implementación de un chatbot que sugiere precios a los turistas representa un avance significativo en la lucha contra el abuso de precios en el sector turístico. Este sistema no sólo proporciona a los visitantes información precisa sobre el costo justo de productos y servicios, sino que también empodera a los turistas para que tomen decisiones informadas. Al tener acceso a una referencia confiable.

Además, este chatbot puede fomentar una mayor equidad en la economía local al reducir las disparidades entre los precios ofrecidos a turistas y locales. Al incentivar a los vendedores a mantener precios competitivos y razonables, se promueve una relación más saludable entre los comerciantes y los visitantes. En última instancia, esta herramienta no solo beneficia a los turistas, sino que también puede resultar en una reputación más positiva para los destinos, atrayendo a más visitantes y promoviendo un turismo sostenible

## REFERENCIAS BIBLIOGRÁFICAS

Jain A.K. Data clustering: 50 years beyond K-means (2010) DOI: 10.1016/j.patrec.2009.09.01. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-77950369345&doi=10.1016%2fj.patrec.2009.09.011&partnerID=40&md5=07fd8ca08e7f0fda5f0ede842575c61>.

Capó M., Pérez A., Lozano J.A. An efficient K-means clustering algorithm for tall data (2020) Data Mining and Knowledge Discovery, 34 (3), pp. 776 - 811. DOI: 10.1007/s10618-020-00678-9. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85081886660&doi=10.1007%2fs10618-020-00678-9&partnerID=40&md5=0bb979130b48bd99d2c66c2c8b6cecc4>

Syakur M.A., Khotimah B.K., Rochman E.M.S., Satoto B.D. Integration K-Means Clustering Method and Elbow Method for Identification of the Best Customer Profile Cluster. DOI: 10.1088/1757-899X/336/1/012017. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85046287552&doi=10.1088%2f1757-899X%2f336%2f1%2f012017&partnerID=40&md5=ef773be5853e1ccfa021375894c727e0>.

Vergara-Schmalbach J.C., Maza-Avila F.J., Martinez-Nagle O., Girado-Guzmán C.A. Evaluation of the quality of the tourist service offered to foreign tourists in the City of Cartagena de Indias, Colombia (2021). DOI: 10.20867/THM.27.2.4. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85111481718&doi=10.20867%2fTHM.27.2.4&partnerID=40&md5=5cafb044ee211b338262f35c9c7c2a09>

► TALENTO  
TECH



Universidad  
Tecnológica  
de Bolívar

[www.utb.edu.co/talento-tech](http://www.utb.edu.co/talento-tech)