

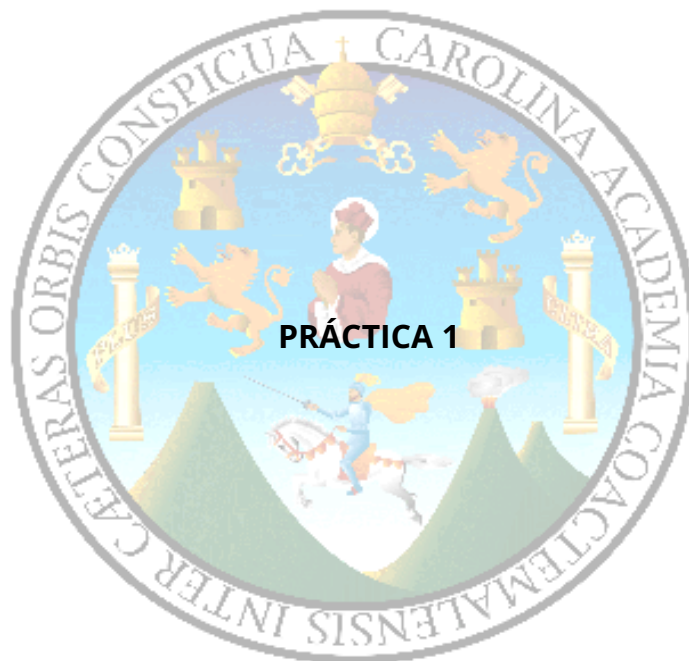
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

ESCUELA DE CIENCIAS Y SISTEMAS

SISTEMAS ORGANIZACIONALES Y GERENCIALES 2

SECCIÓN N°



202004725	Yeinny Melissa Catalán de León
202010223	Luis Angel Barrera Velásquez

Índice

División de Tareas.....	4
Herramientas y Tecnologías.....	4
Establecimiento de Fases del proyecto.....	4
Limpieza y preparación de Datos.....	4
Análisis Exploratorio de Datos.....	4
Desafíos Durante el Análisis.....	4
Selección de Visualizaciones.....	5

PLANIFICACIÓN

División de Tareas

Tarea	Luis Angel Barrera Velásquez	Yeinny Melissa Catalán de León
Preparación de Datos	✓	
Análisis Exploratorio	✓	
Análisis de Tendencias	✓	
Segmentación de Clientes		✓
Análisis de Correlación		✓
Visualización de Datos		✓
Conclusiones y Recomendaciones	✓	✓
Respuestas a las Preguntas	✓	✓

Herramientas y Tecnologías

Base de Datos

- **Base de Datos Utilizada:** MySQL
- **Plataforma en la Nube:** Google Cloud Platform (GCP)

Lenguaje de Programación

- **Lenguaje Principal:** Python

Entorno de Desarrollo

- **Entorno de Desarrollo:** Jupyter Notebook

Librerías y Paquetes Utilizados

- **Pandas:** Para la manipulación y análisis de datos.
- **Matplotlib:** Para la creación de gráficos y visualizaciones.
- **Seaborn:** Para la creación de gráficos estadísticos y visualizaciones más atractivas.
- **MySQL Connector/Python:** Para conectar y operar con la base de datos MySQL.

Herramientas de Visualización

- **Matplotlib:** Utilizado para gráficos básicos, como gráficos de barras y líneas.
- **Seaborn:** Utilizado para gráficos más avanzados, como gráficos de dispersión y distribuciones.

Justificación:

- **MySQL** fue elegido debido a su simplicidad y por su compatibilidad con python.
- **Google Cloud Platform (GCP)** se seleccionó por su escalabilidad e integración con otras herramientas, por lo cual se creó una máquina virtual con docker para almacenar la base de datos.
- **Python** es un lenguaje versátil y ampliamente utilizado en el análisis de datos por su variedad de librerías.
- **Jupyter Notebook** proporcionó un entorno interactivo y flexible para el desarrollo y documentación del análisis.
- **Pandas** facilitó el manejo del archivo csv.
- **Matplotlib** y **Seaborn** permitieron la creación de visualizaciones detalladas y comprensibles.

Establecimiento de Fases del proyecto

1. Preparación y Planificación (31 de agosto de 2024)

- **Objetivo:** Definir el alcance del proyecto, asignar roles y responsabilidades, y establecer un cronograma de trabajo.
- **Actividades:**
 - Revisión del archivo CSV y definición de tareas.
 - Planificación de la estructura de la base de datos y herramientas necesarias.
 - Establecimiento de plazos para cada fase del proyecto.

2. Preparación de Datos (1 - 3 de septiembre de 2024)

- **Objetivo:** Limpiar y preparar los datos para el análisis.
- **Actividades:**
 - Extracción de datos del archivo CSV.
 - Verificación y manejo de valores faltantes y duplicados.
 - Ajuste de tipos de datos y carga de datos en la base de datos MySQL en GCP.

3. Análisis Exploratorio (4 - 6 de septiembre de 2024)

- **Objetivo:** Realizar un análisis inicial de los datos para identificar patrones y obtener estadísticas básicas.
- **Actividades:**
 - Cálculo de estadísticas básicas (media, mediana, moda) para las variables numéricas.
 - Creación de visualizaciones para mostrar la distribución de ventas por categoría de producto y región.

4. Análisis de Tendencias (7 - 8 de septiembre de 2024)

- **Objetivo:** Identificar las tendencias en las ventas y el comportamiento de los clientes.
- **Actividades:**
 - Determinación de los meses con mayores y menores ventas.
 - Identificación de los productos más vendidos y los menos populares.

5. Segmentación de Clientes (9 - 10 de septiembre de 2024)

- **Objetivo:** Agrupar y analizar el comportamiento de los clientes según características demográficas.
- **Actividades:**
 - Agrupación de clientes por edad y análisis de patrones de compra.
 - Comparación del comportamiento de compra entre géneros.

6. Análisis de Correlación (11 de septiembre de 2024)

- **Objetivo:** Investigar relaciones entre diferentes variables en los datos.
- **Actividades:**
 - Investigación de la relación entre el total de la orden y la edad del cliente.
 - Análisis de la correlación entre la categoría del producto y el método de pago preferido.

7. Visualización de Datos (12 de septiembre de 2024)

- **Objetivo:** Crear visualizaciones para representar los hallazgos más importantes del análisis.
- **Actividades:**
 - Creación de gráficos diversos para ilustrar los resultados del análisis.

8. Conclusiones y Recomendaciones (13 de septiembre de 2024)

- **Objetivo:** Sintetizar los hallazgos del análisis y proporcionar recomendaciones para la empresa.
- **Actividades:**


- Redacción de conclusiones clave sobre las ventas y el comportamiento de los clientes.
- Sugerencia de acciones concretas para mejorar las ventas y la satisfacción del cliente.

PROCESO DE ANÁLISIS

Limpieza y preparación de Datos

1. Cargar los datos del archivo CSV

- Se carga el archivo CSV que contiene los datos de ventas de una tienda online.
- `pd.read_csv()` es un método de pandas que lee el archivo y lo convierte en un DataFrame, una estructura de datos tabular.

```
src >  preparacion.py > ...  
1  import pandas as pd  
2  import mysql.connector  
3  
4  # Cargar los datos del archivo CSV  
5  df = pd.read_csv('src/ventas_tienda_online.csv')  
6
```

2. Verificación de valores faltantes y duplicados

- Valores faltantes: Se verifica si alguna columna tiene valores nulos o faltantes usando el método `isnull()` seguido de `sum()`.
- Duplicados: Se verifica si hay filas duplicadas usando el método `duplicated()` y se cuenta la cantidad de duplicados.

```
7  # Verificar valores faltantes  
8  print("Valores faltantes antes de la limpieza:\n", df.isnull().sum())  
9  
10 # Verificar duplicados  
11 print("Duplicados antes de la limpieza: ", df.duplicated().sum())  
12
```

3. Eliminación de duplicados y filas con valores faltantes

- Eliminar duplicados: `drop_duplicates()` elimina las filas duplicadas del DataFrame.
- Eliminar valores faltantes: `dropna()` elimina las filas que contienen valores faltantes (nulos).

```
13  # Eliminar duplicados
14  df = df.drop_duplicates()
15
16  # Eliminar filas con valores faltantes
17  df = df.dropna()
18
```

4. Verificación después de la limpieza

- Después de la limpieza, se vuelve a verificar si hay valores faltantes y duplicados.

```
19  # Verificar nuevamente después de la limpieza
20  print("Valores faltantes después de la limpieza:\n", df.isnull().sum())
21  print("Duplicados después de la limpieza: ", df.duplicated().sum())
22
```

5. Conversión de tipos de datos

- Se asegura que cada columna tenga el tipo de dato correcto:
- `order_id`, `customer_id`, `customer_age`, `quantity` se convierten a enteros.
- `purchase_date`` se convierte a formato de fecha.
- `product_price` y `order_total` se convierten a flotantes (decimales).
- Las demás columnas se convierten en cadenas de texto (str).


```

23 # Asegurar que los tipos de datos son correctos
24 df['order_id'] = df['order_id'].astype(int)
25 df['purchase_date'] = pd.to_datetime(df['purchase_date'])
26 df['customer_id'] = df['customer_id'].astype(int)
27 df['customer_gender'] = df['customer_gender'].astype(str)
28 df['customer_age'] = df['customer_age'].astype(int)
29 df['product_category'] = df['product_category'].astype(str)
30 df['product_name'] = df['product_name'].astype(str)
31 df['product_price'] = df['product_price'].astype(float)
32 df['quantity'] = df['quantity'].astype(int)
33 df['order_total'] = df['order_total'].astype(float)
34 df['payment_method'] = df['payment_method'].astype(str)
35 df['shipping_region'] = df['shipping_region'].astype(str)
36
37 # Verificar que los tipos de datos se han convertido correctamente
38 print(df.dtypes)
39

```

6. Verificación de tipos de datos

Se imprime el tipo de dato de cada columna para asegurarse de que la conversión se realizó correctamente.

```

37 # Verificar que los tipos de datos se han convertido correctamente
38 print(df.dtypes)
39

```

7. Conectar a la base de datos MySQL en GCP

Se establece la conexión con una base de datos MySQL en Google Cloud Platform utilizando la librería mysql.connector.

```

40 # Conectar a la base de datos MySQL en GCP
41 conexion = mysql.connector.connect(
42     host="34.172.242.238",
43     user="root",
44     password="gerenciales13",
45     database="practica13"
46 )

```

8. Truncar la tabla

Se utiliza la consulta TRUNCATE TABLE para eliminar todos los registros de la tabla ventas antes de insertar nuevos datos.

```
50 # Truncar la tabla antes de insertar los nuevos datos
51 cursor.execute("TRUNCATE TABLE ventas")
52
```

9. Preparar la consulta SQL

Se prepara la consulta SQL de inserción para agregar los datos a la tabla ventas. Se utilizan marcadores %s para representar los valores que se van a insertar.

```
53 # Preparar la consulta SQL para insertar los datos en la tabla
54 sql = """
55 INSERT INTO ventas (id_orden, fecha_compra, id_cliente, genero_cliente, edad_cliente,
56                     categoria_producto, nombre_producto, precio_producto, cantidad_comprada,
57                     total_orden, metodo_pago, region_envio)
58 VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
59 """
60
```

10. Convertir los datos a una lista de tuplas

- Se convierte cada fila del DataFrame en una tupla para que los datos puedan ser insertados en la base de datos en un solo lote.
- df.iterrows() permite iterar sobre cada fila del DataFrame.

```
61 # Convertir los datos en una lista de tuplas para la inserción por lotes
62 data = [
63     (
64         row['order_id'],
65         row['purchase_date'],
66         row['customer_id'],
67         row['customer_gender'],
68         row['customer_age'],
69         row['product_category'],
70         row['product_name'],
71         row['product_price'],
72         row['quantity'],
73         row['order_total'],
74         row['payment_method'],
75         row['shipping_region']
76     )
77     for _, row in df.iterrows()
78 ]
```

11. Inserción de los datos en la base de datos

`executemany()` ejecuta la consulta de inserción para cada tupla de datos en la lista `data`.

```
79  
80     # Insertar todos los datos en un solo lote  
81     cursor.executemany(sql, data)  
82
```

12. Confirmación de los cambios

`commit()` guarda los cambios realizados en la base de datos.

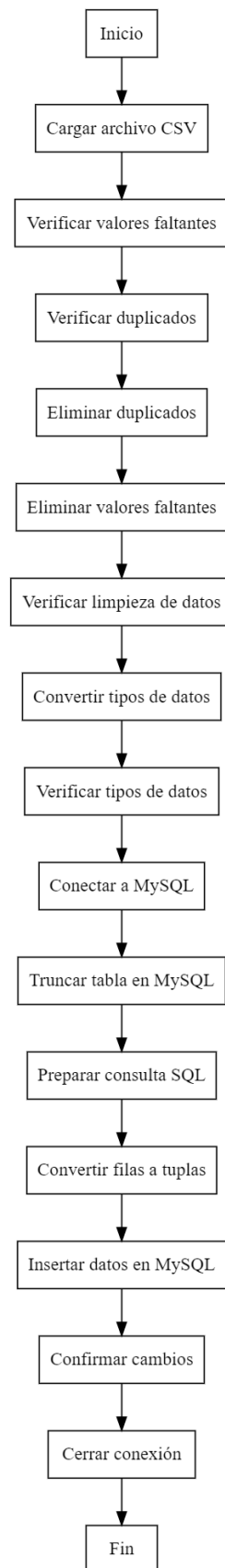
```
83     # Confirmar los cambios en la base de datos  
84     conexion.commit()
```

13. Cerrar la conexión

Se cierran el cursor y la conexión con la base de datos para liberar recursos.

```
87     cursor.close()  
88     conexion.close()  
89
```

Diagrama de flujo sobre el proceso de limpieza y preparación de datos.



Análisis Exploratorio de Datos

1. Selección de Datos Relevantes para el Análisis

Decisión: Elegir las columnas específicas para el análisis a partir de la consulta SQL ejecutada.

Explicación: Se decidió extraer un conjunto específico de columnas de la base de datos (`id_orden`, `fecha_compra`, `id_cliente`, `genero_cliente`, etc.) para cargar en el `DataFrame`. Esta decisión es crucial porque asegura que solo los datos relevantes para el análisis sean incluidos, lo que evita la sobrecarga de información y facilita un análisis más enfocado.

2. Conversión de Tipos de Datos

Decisión: Convertir las columnas `total_orden` y `precio_producto` a valores numéricos.

Explicación: Después de cargar los datos, se decidió convertir las columnas `total_orden` y `precio_producto` a valores numéricos usando `pd.to_numeric()`. Esta decisión es esencial porque asegura que las operaciones matemáticas y el análisis estadístico posterior sean precisos. La conversión a numérico es especialmente importante para evitar errores de tipo de datos que podrían ocurrir si las columnas se mantuvieran como texto.

3. Generación de Estadísticas Básicas y Visualizaciones

Decisión: Calcular estadísticas básicas (media, mediana, moda) para las columnas numéricas y crear visualizaciones para entender la distribución de ventas.

Explicación: Se decidió calcular estadísticas básicas como la media, mediana y moda para todas las columnas numéricas del `DataFrame`. Esta decisión te permite obtener

una visión general de las tendencias y patrones en los datos. Además, elegiste visualizar la distribución de ventas por categoría de producto y región usando un gráfico de barras con seaborn. Esta visualización ayuda a identificar patrones y tendencias en los datos de ventas de manera clara y comprensible, lo que es crucial para tomar decisiones informadas y entender mejor la información.

Desafíos Durante el Análisis

1. Problemas de Conexión con la Base de Datos en GCP

Desafío: El principal desafío que se presentó fue la dificultad para conectar con la base de datos MySQL alojada en Google Cloud Platform (GCP) debido a problemas de configuración del firewall. Esto impidió el acceso inicial a los datos y requirió ajustes en las reglas de red para permitir la conexión desde la máquina local.

Solución: Se revisaron y ajustaron las configuraciones del firewall en GCP para permitir la conexión desde la dirección IP utilizada. Este ajuste resolvió el problema de conexión y permitió continuar con el análisis de datos.

2. Dificultades en el Ordenamiento de Datos en Tablas

Desafío: Otra dificultad encontrada fue el ordenamiento de los datos al cargarlos en el DataFrame. La organización y estructuración de los datos en tablas a veces resultó complicada, especialmente al intentar manejar grandes volúmenes de información y asegurar que las columnas estuvieran correctamente alineadas.

Solución: Se llevaron a cabo varias verificaciones para asegurar que los datos fueran cargados y ordenados correctamente en el DataFrame. Se utilizaron herramientas de Pandas para ajustar y organizar las columnas de manera que se mantuviera la integridad de la estructura de los datos y se facilitarían los análisis posteriores.

METODOLOGÍA

Selección de Visualizaciones

Mejorar y Optimizar los procesos de Ecommerce de la empresa, desde la gestión de productos hasta las relaciones con proveedores y clientes

PREGUNTAS

1. ¿Cómo podrían los insights obtenidos ayudar a diferenciarse de la competencia?

- a. La visualización de las ventas por categoría de producto y región permite identificar qué productos tienen más demanda en cada área. Esto puede guiar la focalización de esfuerzos de marketing en regiones específicas, orientándose a la categoría de productos que son de preferencia en esa región.
- b. Se podrían realizar promociones dependiendo de la categoría que más consuma el sector objetivo, como ejemplo una promoción para el día de la mujer de 2 x 1 en el sector de ropa que es el que más consume esta agrupación.

2. ¿Qué decisiones estratégicas podrían tomarse basándose en este análisis para aumentar las ventas y la satisfacción del cliente?

- a. Se podría implementar que los encargados de marketing se enfoquen en promocionar los productos según el nivel de interés detectado por regiones esto ayudará que si por ejemplo focalizar los esfuerzos de marketing de ropa en la región de Centro y Norte, de accesorios en Norte y Oeste y de calzado en Oeste y Este.
- b. La identificación de los productos más vendidos y los menos populares permiten optimizar el inventario, enfocándose en asegurar un suministro adecuado de los más demandados y reevaluando la estrategia para los

menos populares, mejorando así la disponibilidad y satisfacción del cliente.

3. ¿Cómo podría este análisis de datos ayudar a la empresa a ahorrar costos o mejorar la eficiencia operativa?

- a. El análisis de las tendencias de ventas por mes puede ayudar a planificar la gestión de inventarios. Sabiendo cuándo las ventas tienden a ser más bajas, la empresa puede reducir costos de almacenamiento ajustando la producción y la logística.
- b. Identificar los productos menos populares ayuda a reducir costos al evitar el almacenamiento excesivo de inventario innecesario, permitiendo que los recursos sean invertidos en productos de mayor demanda.

4. ¿Qué datos adicionales recomendarían recopilar para obtener insights aún más valiosos en el futuro?

- a. Sería bueno recopilar información de los clientes después de las compras realizadas para poder recibir retroalimentación del nivel de satisfacción que reciben al adquirir los productos y así evaluar si se debe descontinuar o no un producto.
- b. Datos sobre la frecuencia y valor promedio de las devoluciones o reembolsos proporcionarían información clave sobre la calidad de los productos y el servicio, ayudando a identificar áreas de mejora para la eficiencia operativa y la experiencia del cliente.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- El análisis de los datos de ventas revela que la categoría de ropa constituye la principal fuente de ingresos para la empresa, superando en rentabilidad a las categorías de calzado y accesorios, tanto en el segmento masculino como femenino.

- El análisis revela que los segmentos de edad más jóvenes (18-24 y 25-34 años) presentan los índices de compra más elevados, tanto en términos de frecuencia como de ticket promedio, lo que indica un mayor nivel de engagement con la marca por parte de este grupo demográfico.
- El gráfico muestra una distribución bastante equilibrada entre los cuatro métodos de pago principales: transferencia bancaria, tarjeta de crédito, efectivo y PayPal. Ningún método destaca significativamente sobre los demás, lo que sugiere que la empresa ofrece una variedad de opciones de pago que se adaptan a las preferencias de diferentes segmentos de clientes.
- La empresa ha experimentado un crecimiento constante en su base de clientes o en la demanda de sus productos/servicios durante el período analizado. Esto sugiere que la estrategia comercial implementada está dando resultados positivos.

Recomendaciones

- Lanzar promociones enfocadas en las categorías de productos más consumidos por región. Por ejemplo, ofrecer descuentos en ropa en zonas donde este sea el producto más demandado. Esto maximizaría el impacto del marketing y aumentaría la satisfacción al ofrecer productos relevantes a cada mercado.
- Ajustar el inventario enfocándose en los productos más vendidos y realizando liquidaciones de los menos populares. Esto reduciría costos de almacenamiento, mejoraría la eficiencia operativa y garantizaría la disponibilidad de productos clave, mejorando la experiencia del cliente.
- Utilizar datos de compras anteriores para crear campañas de retargeting dirigidas a clientes que han mostrado interés en ciertas categorías de productos pero no han completado una compra.
- Implementar encuestas de satisfacción del cliente después de cada compra para recoger información sobre la experiencia del cliente y áreas de mejora.