



Unlocking OLIST Potential -

Insight from Customer, Sales, and Review Data

Final Project of Data Analyst (Batch 28) Course at Growia





Table of Contents

1. Background

2. The Objective

3. Data Explanation

4. Analysis

**5. Conclusions &
Recommendations**

Background

Olist adalah platform e-commerce terkemuka di Brazil yang menyediakan solusi lengkap untuk bisnis online. Dataset Olist, yang berisi data transaksi e-commerce yang komprehensif, menawarkan wawasan berharga tentang perilaku konsumen dan dinamika pasar Brazil.

Analisis ini bertujuan untuk menggali informasi penting dari dataset Olist, termasuk segmentasi pelanggan, tren penjualan produk, preferensi pembayaran, dan kepuasan pelanggan. Dengan pemahaman yang lebih dalam tentang faktor-faktor ini, pelaku bisnis di platform Olist dapat membuat keputusan yang lebih tepat, meningkatkan efisiensi operasional, dan dapat mendorong pertumbuhan dan profitabilitas di pasar e-commerce Brazil yang kompetitif.



The Objectives

Objective 01

Memahami profil dan segmentasi pelanggan Olist untuk mengidentifikasi pasar yang potensial

Objective 02

Mengenali tren dan pola penjualan produk untuk mengoptimalkan strategi pemasaran dan inventaris

Objective 03

Mengetahui tingkat kepuasan pelanggan dan faktor-faktor yang mempengaruhinya untuk meningkatkan pengalaman pelanggan dan membangun loyalitas

Data Explanation

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 119143 entries, 0 to 119142
```

```
Data columns (total 40 columns):
```

#	Column	Non-Null Count	Dtype
0	product_id	118310 non-null	object
1	product_category_name	116576 non-null	object
2	product_name_lenght	116576 non-null	object
3	product_description_lenght	116576 non-null	object
4	product_photos_qty	116576 non-null	object
5	product_weight_g	116576 non-null	object
6	product_length_cm	116576 non-null	object
7	product_height_cm	116576 non-null	object
8	product_width_cm	116576 non-null	object
9	product_category_name_english	116576 non-null	object
10	order_id	119143 non-null	object
11	order_item_id	118310 non-null	object
12	seller_id	118310 non-null	object
13	shipping_limit_date	118310 non-null	object
14	price	118310 non-null	object
15	freight_value	118310 non-null	object
16	customer_id	119143 non-null	object
17	order_status	119143 non-null	object
18	order_purchase_timestamp	119143 non-null	object
19	order_approved_at	119143 non-null	object
20	order_delivered_carrier_date	119143 non-null	object
21	order_delivered_customer_date	119143 non-null	object
22	order_estimated_delivery_date	119143 non-null	object
23	customer_unique_id	119143 non-null	object
24	customer_zip_code_prefix	119143 non-null	object
25	customer_city	119143 non-null	object
26	customer_state	119143 non-null	object
27	seller_zip_code_prefix	118310 non-null	object
28	seller_city	118310 non-null	object

29	seller_state	118310 non-null	object
30	payment_sequential	119140 non-null	object
31	payment_type	119140 non-null	object
32	payment_installments	119140 non-null	object
33	payment_value	119140 non-null	object
34	review_id	118146 non-null	object
35	review_score	118146 non-null	object
36	review_comment_title	118146 non-null	object
37	review_comment_message	118146 non-null	object
38	review_creation_date	118146 non-null	object
39	review_answer_timestamp	118146 non-null	object

The source: Growia



olist.sql: olist_customers_dataset,
olist_geolocation_dataset, olist_orders_dataset,
olist_order_items_dataset, olist_order_payments_dataset,
olist_order_reviews_dataset, olist_products_dataset,
olist_sellers_dataset, product_category_name_translation



40 kolom dan 119.143 baris

Data Explanation

```
# Delete unnecessary columns
data_filtered = data_merged.drop(columns=['product_photos_qty', 'product_name_lenght', 'product_description_lenght',
                                           'product_length_cm', 'product_width_cm', 'product_height_cm',
                                           'customer_zip_code_prefix', 'review_answer_timestamp',
                                           'review_creation_date'])

data_filtered.info()
```

Delete Unnecessary Columns

```
# Mengganti empty string di 'product_weight_g' dengan NaN
data_filtered['product_weight_g'] = pd.to_numeric(data_filtered['product_weight_g'], errors='coerce')

# Mengubah tipe data kolom tertentu
data_filtered = data_filtered.astype({
    'order_purchase_timestamp': 'datetime64[ns]',
    'order_approved_at': 'datetime64[ns]',
    'order_delivered_carrier_date': 'datetime64[ns]',
    'order_delivered_customer_date': 'datetime64[ns]',
    'order_estimated_delivery_date': 'datetime64[ns]',
    'payment_sequential': 'float64',
    'payment_installments': 'float64',
    'payment_value': 'float64',
    'review_score': 'float64',
    'price': 'float64',
    'order_item_id': 'float64',
    'shipping_limit_date': 'datetime64[ns]',
    'freight_value': 'float64',
    'product_weight_g': 'float64',
})

# Menampilkan info data_filtered untuk melihat perubahan tipe data
data_filtered.info()
```

Changes Dtypes

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119143 entries, 0 to 119142
Data columns (total 31 columns):
```

#	Column	Non-Null Count	Dtype
0	product_id	118310 non-null	object
1	product_category_name	116576 non-null	object
2	product_weight_g	116575 non-null	float64
3	product_category_name_english	116576 non-null	object
4	order_id	119143 non-null	object
5	order_item_id	118310 non-null	float64
6	seller_id	118310 non-null	object
7	shipping_limit_date	118310 non-null	datetime64[ns]
8	price	118310 non-null	float64
9	freight_value	118310 non-null	float64
10	customer_id	119143 non-null	object
11	order_status	119143 non-null	object
12	order_purchase_timestamp	119143 non-null	datetime64[ns]
13	order_approved_at	118966 non-null	datetime64[ns]
14	order_delivered_carrier_date	117057 non-null	datetime64[ns]
15	order_delivered_customer_date	115722 non-null	datetime64[ns]
16	order_estimated_delivery_date	119143 non-null	datetime64[ns]
17	customer_unique_id	119143 non-null	object
18	customer_city	119143 non-null	object
19	customer_state	119143 non-null	object
20	seller_zip_code_prefix	118310 non-null	object
21	seller_city	118310 non-null	object
22	seller_state	118310 non-null	object
23	payment_sequential	119140 non-null	float64
24	payment_type	119140 non-null	object
25	payment_installments	119140 non-null	float64
26	payment_value	119140 non-null	float64
27	review_id	118146 non-null	object
28	review_score	118146 non-null	float64
29	review_comment_title	118146 non-null	object
30	review_comment_message	118146 non-null	object

```
dtypes: datetime64[ns](6), float64(8), object(17)
```

```
memory usage: 28.2+ MB
```

Data Explanation

Handling Missing Values

```
# Mendapatkan jumlah missing value tiap kolom
nan_col = data_filtered.isna().sum().sort_values(ascending=False)
nan_col

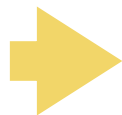
# Mendapatkan persentase missing value tiap kolom
n_data = len(data_filtered)

percent_nan_col = (nan_col / n_data) * 100
percent_nan_col

# Show count dan persen NaN
df_nan_col = pd.DataFrame({'count_nan': nan_col, 'persen_nan': percent_nan_col})
df_nan_col
```

```
# Drop missing values
data_clean = data_filtered.dropna()

# Check % apakah masih ada missing values
check_nan_col = data_clean.isna().sum() / len(data_clean) * 100
check_nan_col
```



	count_nan	persen_nan
order_delivered_customer_date	3421	2.871339
product_weight_g	2568	2.155393
product_category_name_english	2567	2.154554
product_category_name	2567	2.154554
order_delivered_carrier_date	2086	1.750837
review_comment_title	997	0.836810
review_score	997	0.836810

product_id	0.0	order_purchase_timestamp	0.0
product_category_name	0.0	order_approved_at	0.0
product_weight_g	0.0	order_delivered_carrier_date	0.0
product_category_name_english	0.0	order_delivered_customer_date	0.0
order_id	0.0	order_estimated_delivery_date	0.0
order_item_id	0.0	customer_unique_id	0.0
seller_id	0.0	customer_city	0.0
shipping_limit_date	0.0	customer_state	0.0

Data Explanation

```
# Add time column based on order_approved dates
data_clean['date'] = pd.to_datetime(data_clean['order_purchase_timestamp'])
data_clean['day'] = data_clean['date'].dt.day
data_clean['month'] = data_clean['date'].dt.month
data_clean['year'] = data_clean['date'].dt.year
data_clean['hour'] = data_clean['date'].dt.hour
data_clean['year_month'] = data_clean['date'].dt.to_period("M")
```

```
# Check new time columns
data_clean[['date', 'year', 'month', 'day', 'hour', 'year_month']].head(3)
```

	date	year	month	day	hour	year_month
0	2017-09-13 08:59:02	2017	9	13	8	2017-09
1	2017-04-26 10:53:06	2017	4	26	10	2017-04
2	2018-01-14 14:33:31	2018	1	14	14	2018-01

Add Columns

```
data_clean['total_order_value'] = data_clean['price'] + data_clean['freight_value']
```

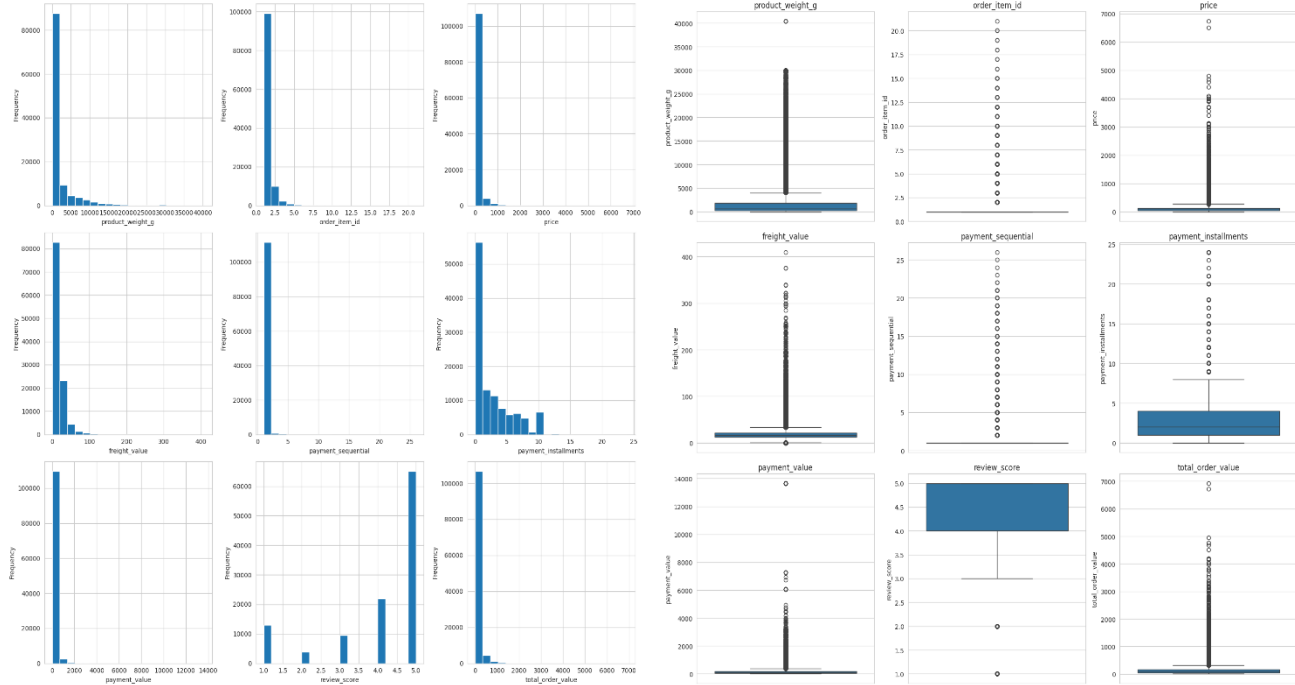
```
# show result
data_clean[['order_id', 'year_month', 'price', 'freight_value', 'total_order_value']].sample(3)
```

	order_id	year_month	price	freight_value	total_order_value
105407	e20048c62eafbe51818f6dd9f3be724e	2018-03	49.9	8.40	58.30
95689	cd63595a31d3717ff7ea9a0b041104a9	2017-01	24.9	14.52	39.42
39459	54497c0e9ae7aa2ffe4bbb29883d59cb	2017-08	39.9	16.79	56.69

Exploratory Data Analysis

Histogram dan Boxplot digunakan untuk mengidentifikasi tren dan outlier dalam kumpulan data.

Secara keseluruhan, histogram dan boxplot memberikan gambaran yang konsisten tentang distribusi data dalam **data_clean**. Sebagian besar kolom numerik memiliki distribusi right-skewed dengan outlier, menunjukkan adanya nilai-nilai ekstrem yang perlu dipertimbangkan dalam analisis.

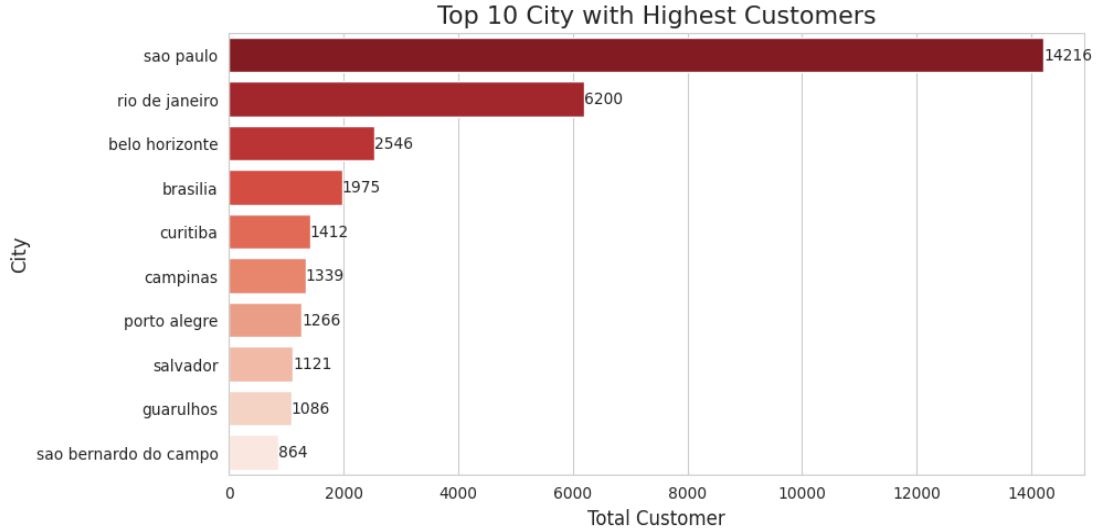


Exploratory Data Analysis (1)

Top City with Highest Customers

Sao Paulo dan Rio de Janeiro

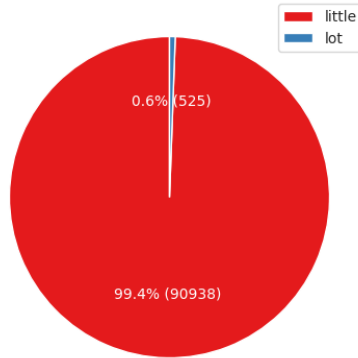
menunjukkan perbedaan signifikan dalam jumlah pelanggan. Sao Paulo mendominasi dengan 14.216 pelanggan, lebih dari dua kali lipat jumlah pelanggan di Rio de Janeiro yang memiliki 6.200 pelanggan.



Exploratory Data Analysis

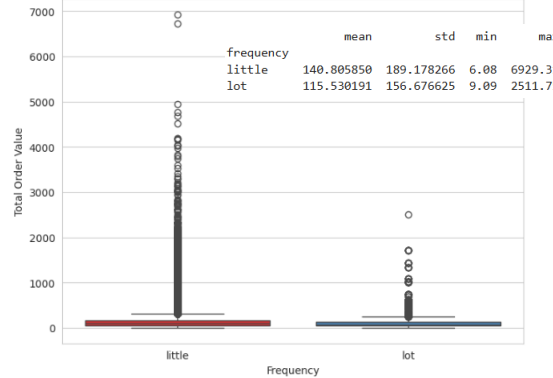
Customer Frequency Segmentation

Order Frequency Segmentation

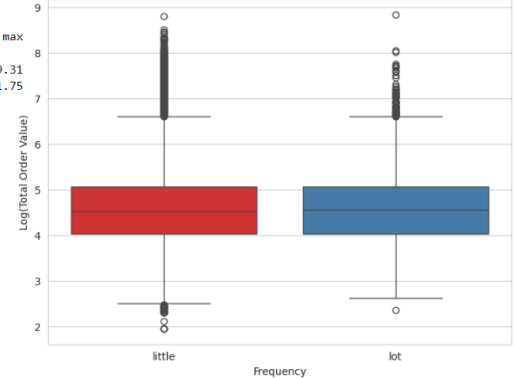


Sebagian besar customer termasuk dalam segmen **"little"**. Jumlah customer pada segmen **"lot"** jauh lebih sedikit.

Distribution of Total Order Value by Frequency



Distribution of Total Order Value by Frequency (Log Transformed)



Terlihat jelas bahwa kelompok "little" memiliki median nilai order yang lebih tinggi dan variabilitas yang lebih besar dibandingkan kelompok "lot". Selain itu, terlihat beberapa outlier pada kedua kelompok, terutama pada kelompok "little" dengan nilai order yang sangat tinggi.

Exploratory Data Analysis

Distribution Order Frequency by Price Classify

Kategori "**cheap**" cenderung memiliki frekuensi order tertinggi, diikuti oleh "**normal**", dan terakhir "**expensive**".

Terdapat variasi yang signifikan pada rata-rata total order antar kategori harga. Ini menunjukkan bahwa harga produk memiliki pengaruh besar terhadap nilai total order.

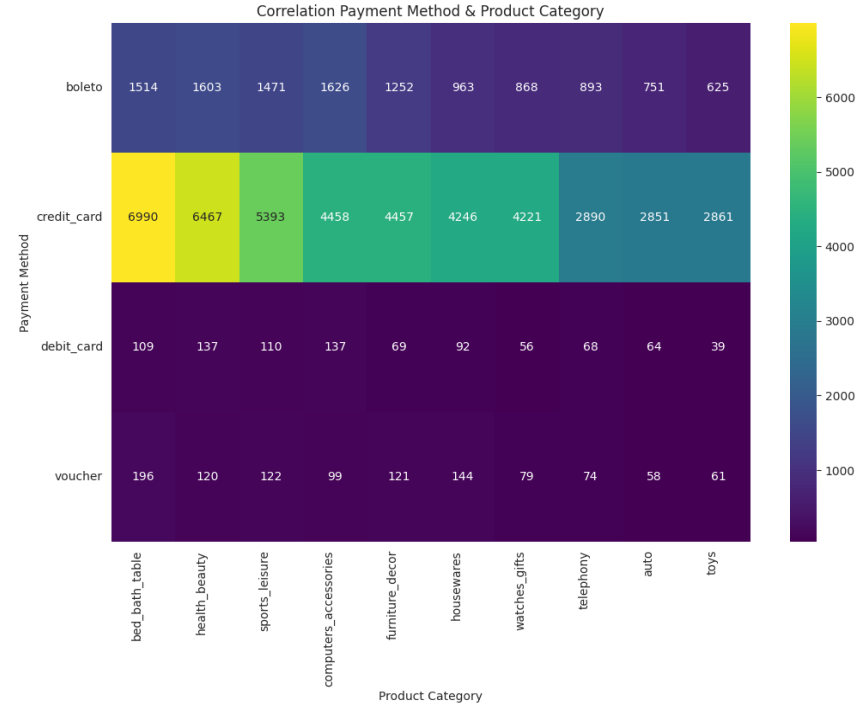


Exploratory Data Analysis

Correlation Payment Method & Product Category

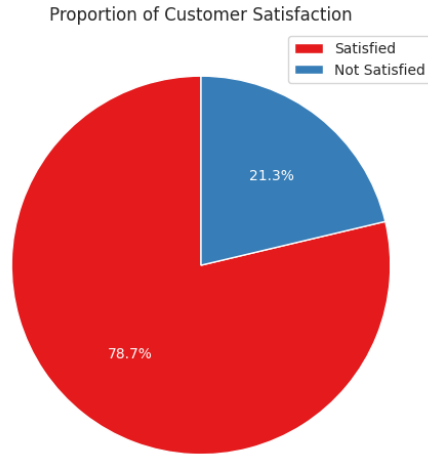
Heatmap ini memvisualisasikan hubungan antara metode pembayaran dan preferensi produk. Beberapa poin penting yang dapat disimpulkan:

- **Credit Card** adalah metode pembayaran yang paling populer.
- Kategori produk yang paling banyak dibeli adalah **bed_bath_table**.
- Metode pembayaran lain seperti **boleto**, **debit_card**, dan **voucher** kurang populer.

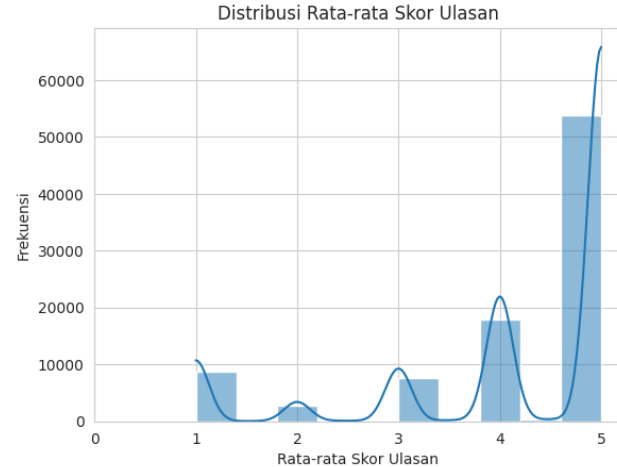


Exploratory Data Analysis

Customer Satisfaction



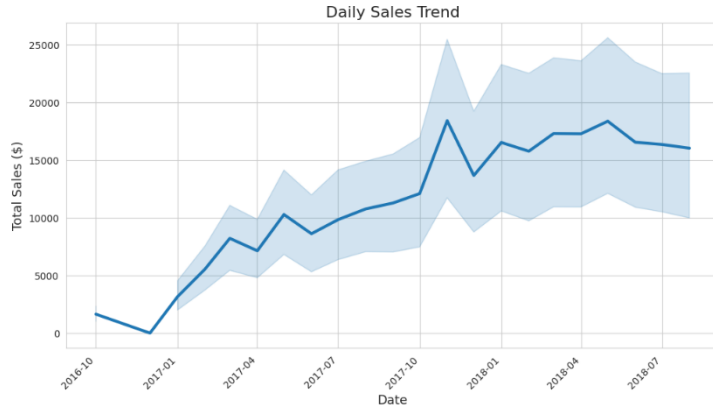
Mayoritas customer (78.7%) merasa puas, sementara 21.3% lainnya tidak puas.



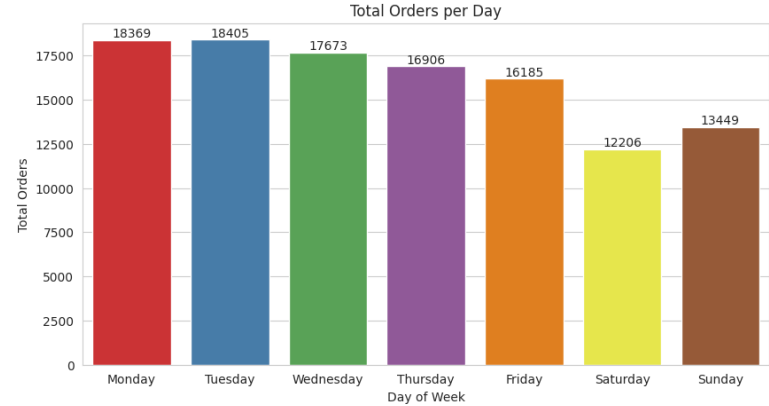
Menampilkan distribusi **review score**, dengan frekuensi tertinggi pada skor 5 dan penurunan bertahap pada skor 4,3 dan seterusnya.

Exploratory Data Analysis

Time Series Analysis (DoD)



- Tren penjualan harian dari Oktober 2016 hingga Juli 2018, menunjukkan peningkatan umum dengan fluktuasi musiman.
- Variabilitas penjualan juga terlihat, dengan beberapa titik data menonjol yang memerlukan investigasi lebih lanjut.



Grafik ini memperlihatkan fluktuasi jumlah pesanan harian dalam seminggu, dengan volume tertinggi pada hari Senin dan Selasa, diikuti penurunan bertahap hingga akhir pekan.

Conclusion

1. **Pelanggan dan Pasar:** Sebagian besar pelanggan Olist berasal dari Sao Paulo dan Rio de Janeiro, dan mereka cenderung melakukan sedikit transaksi. Ini menunjukkan bahwa pasar utama Olist terkonsentrasi di kedua wilayah ini, dan fokus perlu diberikan pada akuisisi pelanggan baru serta meningkatkan frekuensi transaksi pelanggan yang sudah ada.
2. **Preferensi dan Pembayaran:** Pelanggan Olist paling banyak membeli produk dari kategori bed_bath_table, dan mereka mayoritas menggunakan credit card sebagai metode pembayaran. Hal ini menunjukkan pentingnya memastikan ketersediaan stok produk bed_bath_table dan kerja sama yang baik dengan penyedia layanan credit card.
3. **Kepuasan dan Loyalitas:** Tingkat kepuasan pelanggan Olist relatif tinggi, yang ditunjukkan dengan skor ulasan yang baik. Mempertahankan dan meningkatkan kepuasan pelanggan sangat penting untuk membangun loyalitas.
4. **Tren dan Pola Penjualan:** Penjualan Olist cenderung meningkat dari waktu ke waktu, dengan puncaknya di bulan November. Terdapat pula pola penjualan harian, dengan total pesanan tertinggi di hari Senin dan terendah di hari Minggu. Informasi ini dapat digunakan untuk mengoptimalkan strategi pemasaran dan operasional.

Business Recommendations

- 1. Akuisisi dan Retensi Pelanggan:** Fokuskan strategi pemasaran di Sao Paulo dan Rio de Janeiro untuk menjangkau lebih banyak pelanggan potensial. Selain itu, kembangkan program loyalitas atau penawaran khusus untuk mendorong pelanggan yang sudah ada untuk bertransaksi lebih sering.
- 2. Manajemen Produk dan Pembayaran:** Pastikan ketersediaan stok produk bed_bath_table untuk memenuhi permintaan yang tinggi. Pertahankan kerja sama dengan penyedia layanan credit card, dan pertimbangkan untuk menambahkan metode pembayaran lain yang populer untuk memberikan fleksibilitas kepada pelanggan.
- 3. Peningkatan Pengalaman Pelanggan:** Terus tingkatkan kualitas produk dan layanan untuk mempertahankan dan meningkatkan kepuasan pelanggan. Tanggapi masukan dan ulasan pelanggan secara proaktif, dan berusahalah untuk memberikan pengalaman belanja yang positif.
- 4. Optimasi Penjualan dan Operasional:** Manfaatkan tren dan pola penjualan untuk mengoptimalkan strategi pemasaran dan operasional. Tingkatkan promosi di bulan-bulan dengan penjualan rendah, dan optimalkan operasional bisnis di awal minggu untuk menangani jumlah pesanan yang lebih tinggi.

Thanks

Do you have any questions?

helmifitrayadi@gmail.com

<https://www.linkedin.com/in/helmifitrayadi/>



CREDITS: This presentation template was created by [Slidesgo](#), including icons by [Flaticon](#), and infographics & images by [Freepik](#)

Please keep this slide for attribution