FACEBOOK CAMPAIGN ANALYSIS

January 2023

BY: AMR ELLAKANY

amrellakany600@gmail.com

https://amr-ellakany.jimdosite.com/

CONTENT

01 Terminology Defining

02 Dataset Exploring

03 Campaign Highlights

04 Lookalike vs Detailed Targeting

05 Various Funnel Metrics

06 Statistics & Charts

PROJECT'S LINK:
[Click to Explore The Code]

TERMINOLOGY

DEFINING

Go back to Agenda Page

THE MARKETING FUNNEL

01 Awareness

02 Interest

03 Desire

03 Action

AUDIENCE

01 Core (Detailed Targeting)

02 Lookalike

03 Custom (Hot & Warm)

DATASET

EXPLORING

Go back to Agenda Page

THE DATASET

```
In [104]: import pandas as pd
          df = pd.read_csv(r"C:\Users\user\Desktop\FB Campign analysis\A Marketing campaign Dataset.csv")
          df.head()
```

Out[104]:

	Reporting Starts	Reporting Ends	Ad Set Name	Amount Spent	Results	Result indicator	CPR	Reach	Impressions	Frequency	CPM	CPC	CTF
0	20/5/2022	31/5/2022	T1 - Cold - LLA - Promo May	332.65	2	actions:offsite_conversion.fb_pixel_purchase	166.3250	6032	12594	2.087865	26.413371	2.294138	2.2(
1	20/5/2022	31/5/2022	T1 - Cold - DT - Promo May	915.00	15	actions:offsite_conversion.fb_pixel_purchase	61.0000	18900	42510	2.249206	21.524347	2.636888	1.5ε
2	20/5/2022	31/5/2022	T2 - Cold - LLA - Promo May	327.77	4	actions:offsite_conversion.fb_pixel_purchase	81.9425	34743	61268	1.763463	5.349775	0.947312	1.2:
3	20/5/2022	31/5/2022	T2 - Cold - DT - Promo May	297.38	0	NaN	NaN	45001	65880	1.463967	4.513965	0.788806	1.22
4	20/5/2022	31/5/2022	T3 - Cold - LLA - Promo May	291.45	4	actions:offsite_conversion.fb_pixel_purchase	72.8625	69040	147600	2.137891	1.974594	0.464092	1.11

THE DATASET

Ad Set Overview

- Cold, Warm, and Hot ad sets
- lookalike & Detailed Targeting in Cold ad set
- T1, T2, T3, and T4 in Cold Ad set
- CRSL & Videos in Warm Ad set
- Last chance, Discount, and Last day in Hot ad set

```
In [6]: (df['Ad Set Name']).unique()

Out[6]: array(['T1 - Cold - LLA - Promo May', 'T1 - Cold - DT - Promo May',
               'T2 - Cold - LLA - Promo May', 'T2 - Cold - DT - Promo May',
               'T3 - Cold - LLA - Promo May', 'T3 - Cold - DT - Promo May',
               'T4 - Cold - DT - Promo May', 'T4 - Cold - LLA - Promo May',
               'Warm - CRSL', 'Warm - Videos', 'HOT - Last day',
               'HOT - Last chance & crsl', 'HOT - Discount'], dtype=object)
```

CREATING A CATEGORY COLUMN FOR DIFFERENT AD SETS

Hot, Warm, and Cold Ad Sets

In [70]:
```
def categorize_ad_set(ad_set_name):
    if 'Warm' in ad_set_name:
        return 'Warm'
    elif 'HOT' in ad_set_name:
        return 'Hot'
    else:
        return 'Cold'

df['Category'] = df['Ad Set Name'].apply(categorize_ad_set)
df.head()
```

Out[70]:

Amount Spent	Results	Result indicator	CPR	Reach	Impressions	Frequency	CPM	CPC	CTR (all)	Link Clicks	Landing Page Views	Category
332.65	2	actions:offsite_conversion.fb_pixel_purchase	166.3250	6032	12594	2.087865	26.413371	2.294138	2.207400	145	89	Cold
915.00	15	actions:offsite_conversion.fb_pixel_purchase	61.0000	18900	42510	2.249206	21.524347	2.636888	1.587862	347	215	Cold
327.77	4	actions:offsite_conversion.fb_pixel_purchase	81.9425	34743	61268	1.763463	5.349775	0.947312	1.237187	346	220	Cold
297.38	0	NaN	NaN	45001	65880	1.463967	4.513965	0.788806	1.220401	377	225	Cold

CREATING A SUB-CATEGORY COLUMN FOR DIFFERENT AD SETS INSIDE COLD AD SET

Looalike vs Detailed Targeting

In [20]:
```
df['Ad Set Name'].unique()
```

Out[20]:
```
array(['T1 - Cold - LLA - Promo May', 'T1 - Cold - DT - Promo May',
       'T2 - Cold - LLA - Promo May', 'T2 - Cold - DT - Promo May',
       'T3 - Cold - LLA - Promo May', 'T3 - Cold - DT - Promo May',
       'T4 - Cold - DT - Promo May', 'T4 - Cold - LLA - Promo May',
       'Warm - CRSL', 'Warm - Videos', 'HOT - Last day',
       'HOT - Last chance & crsl', 'HOT - Discount'], dtype=object)
```

In [105]:
```
def categorize_ad_set_2(ad_set_name):
    if('DT' in ad_set_name):
        return 'Detailed Targeting'
    elif('LLA' in ad_set_name):
        return 'Lookalike Audience'
    else:
        return 'Other'

df['Subcategory'] = df['Ad Set Name'].apply(categorize_ad_set_2)
df.head()
```

Out[105]:

mount Spent	Results	Result indicator	CPR	Reach	Impressions	Frequency	CPM	CPC	CTR (all)	Link Clicks	Landing Page Views	Subcategory
332.65	2	actions:offsite_conversion.fb_pixel_purchase	166.3250	6032	12594	2.087865	26.413371	2.294138	2.207400	145	89	Lookalike Audience
915.00	15	actions:offsite_conversion.fb_pixel_purchase	61.0000	18900	42510	2.249206	21.524347	2.636888	1.587862	347	215	Detailed Targeting

CAMPAIGN HIGHLIGHTS

Go back to Agenda Page

CAMPAIGN HIGHLIGHTS

- We reached around **450K** unique people.
- Approx **1.1 million** impressions.
- Each person saw the ad around **2.5** times.
- Approx **5.6K** link clicks and **2.9K** landing page views.
- A total of **83** results.
- Spent **4,362 EUR**.

```
In [69]: df[["Reach","Impressions","Link Clicks","Landing Page Views","Results","Amount Spent"]].sum()

Out[69]: Reach                   453534.00
         Impressions            1136052.00
         Link Clicks               5589.00
         Landing Page Views        2985.00
         Results                     83.00
         Amount Spent              4362.12
         dtype: float64
```

AGGREGATING ALL METRICS BASED ON THE CATEGORY

```
In [72]: results = df.groupby('Category').agg({'Reach':'mean','Impressions':'mean',
                                               'CTR (all)':'mean','Results':'mean',
                                               'CPR':'mean'})

         results
```

Out[72]:

Category	Reach	Impressions	CTR (all)	Results	CPR
Cold	44435.375000	79417.500000	1.370799	4.000000	80.352667
Hot	11195.666667	58889.666667	1.034126	10.333333	38.851667
Warm	32232.000000	162021.500000	1.052970	10.000000	51.310357

REACH — **Cold** ad set has the **highest** reach, indicating success in reaching a broad audience.

IMPRESSIONS — **Warm** ad set generates the **highest** impressions, surpassing Cold, **despite** a smaller reach.

CTR — **Cold** Ad sets show **higher** CTR, possibly due to less familiarity with the company's content.

HOT & WARM PERFORMANCE — They perform **the best**, aligning with expectations of higher familiarity and conversion **likelihood**.

CPR — **Hot** Ad set has the **lowest** CPR, indicating **cost-effectiveness**, likely due to a closer audience ready for purchase as they are more familiar.

AGGREGATING ALL METRICS BASED ON THE SUB-CATEGORY

```
In [74]: df_cold = df[df['Category']=='Cold']

In [75]: df_cold = df_cold.groupby('Subcategory').agg({'Reach':'mean','Impressions':'mean',
                                                  'CTR (all)':'mean','Results':'mean',
                                                  'CPR':'mean'})

         df_cold
```

Out[75]:

Subcategory	Reach	Impressions	CTR (all)	Results	CPR
Detailed Targeting	34033.25	53142.25	1.351996	4.25	53.2550
Lookalike Audience	54837.50	105692.75	1.389602	3.75	93.9015

LOOKALIKE VS DETAILED TARGETING

Go back to Agenda Page

LOOKALIKE VS DETAILED TARGETING [UNPIVOTING & PLOTTING]

In [77]:
```
df_transformed = df_cold.reset_index().melt('Subcategory')
df_transformed
```

Out[77]:

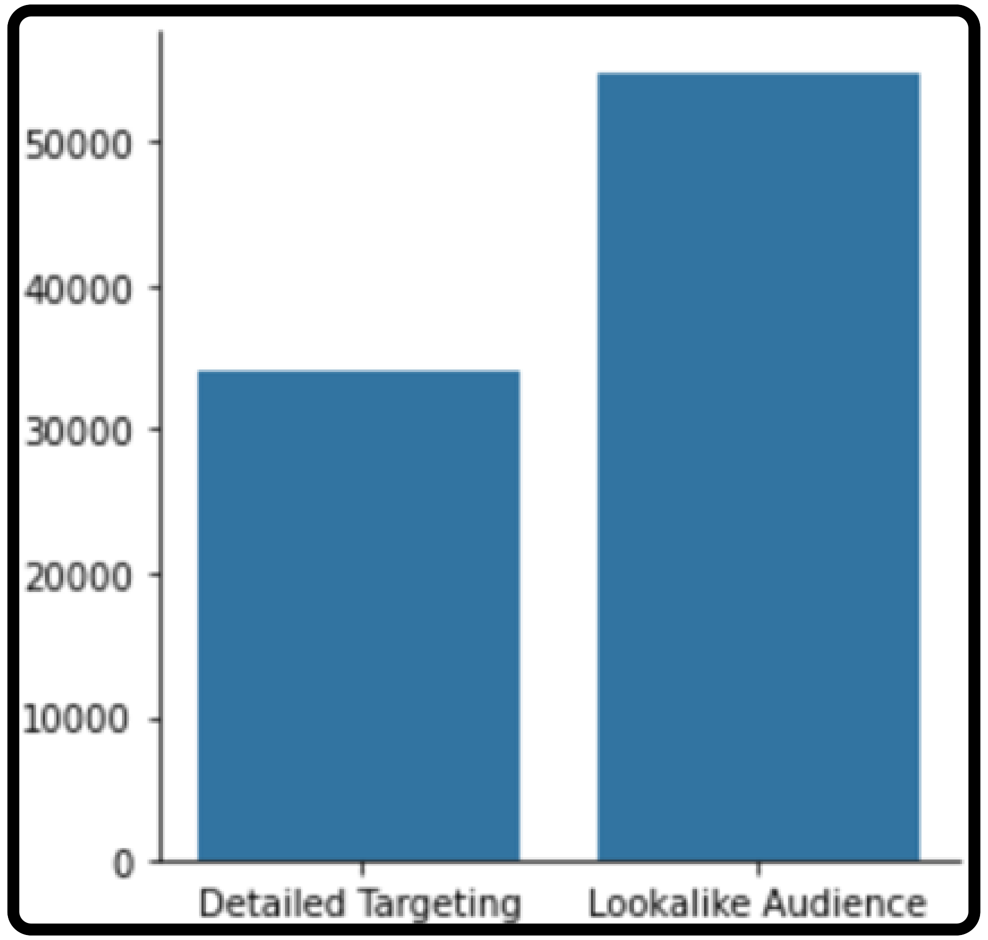
	Subcategory	variable	value
0	Detailed Targeting	Reach	34033.250000
1	Lookalike Audience	Reach	54837.500000
2	Detailed Targeting	Impressions	53142.250000
3	Lookalike Audience	Impressions	105692.750000
4	Detailed Targeting	CTR (all)	1.351996
5	Lookalike Audience	CTR (all)	1.389602
6	Detailed Targeting	Results	4.250000
7	Lookalike Audience	Results	3.750000
8	Detailed Targeting	CPR	53.255000
9	Lookalike Audience	CPR	93.901500

In [83]:
```
viz = sns.FacetGrid(df_transformed, col='variable', col_wrap=3, sharex=False,
                    sharey=False, height=4)

viz.map(sns.barplot, 'Subcategory', 'value', order=df_cold.reset_index()['Subcategory'])

plt.show()
```

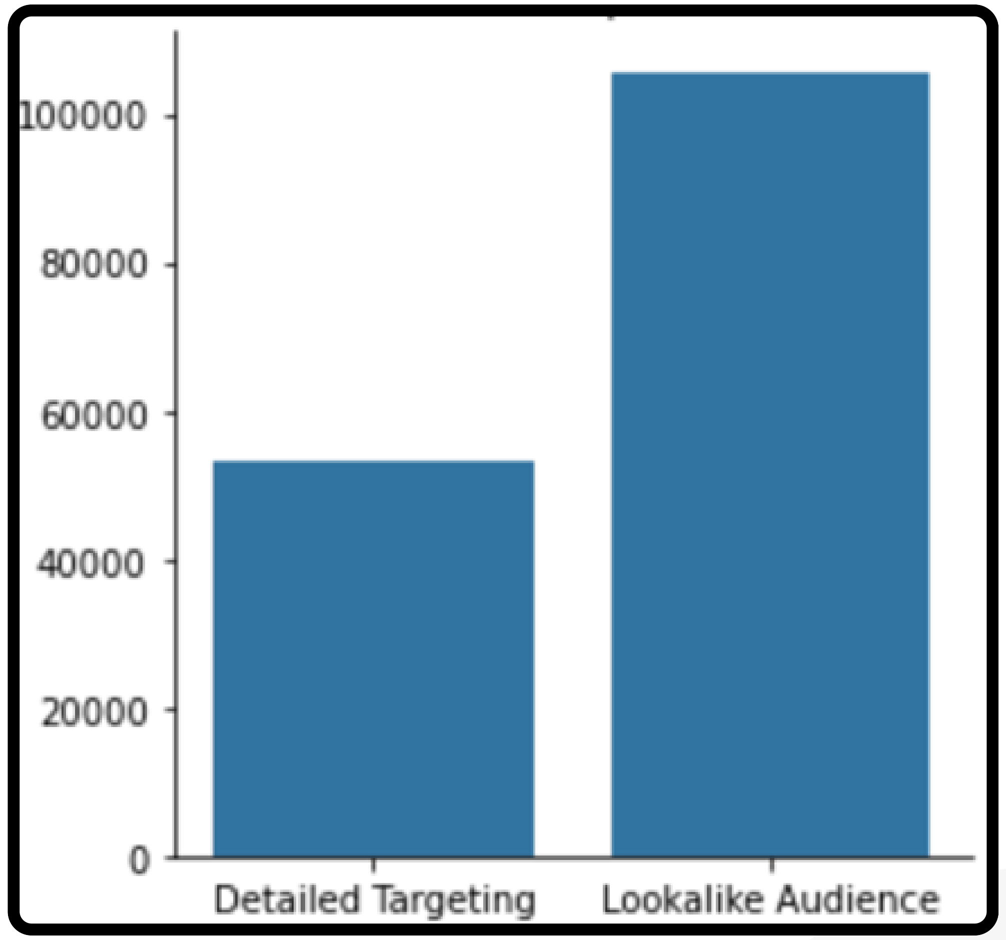
LOOKALIKE VS DETAILED TARGETING [REACH & IMPRESSION]

- **Reach & Impression**

The Lookalike audience has a **higher reach** and **impressions** compared to the Detailed audience, as Lookalikes resemble a similar base as the existing audience.

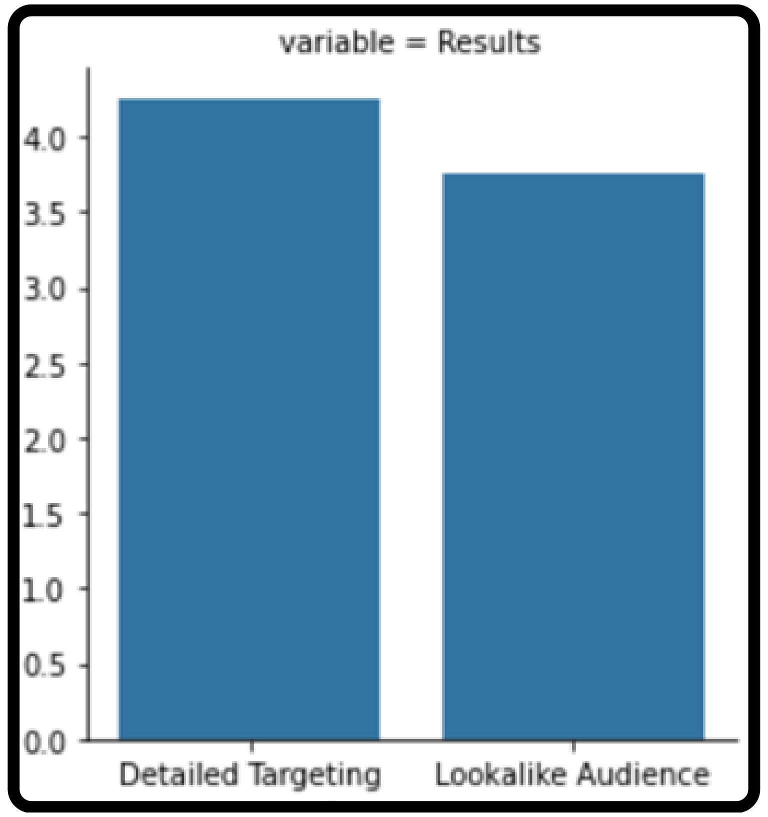


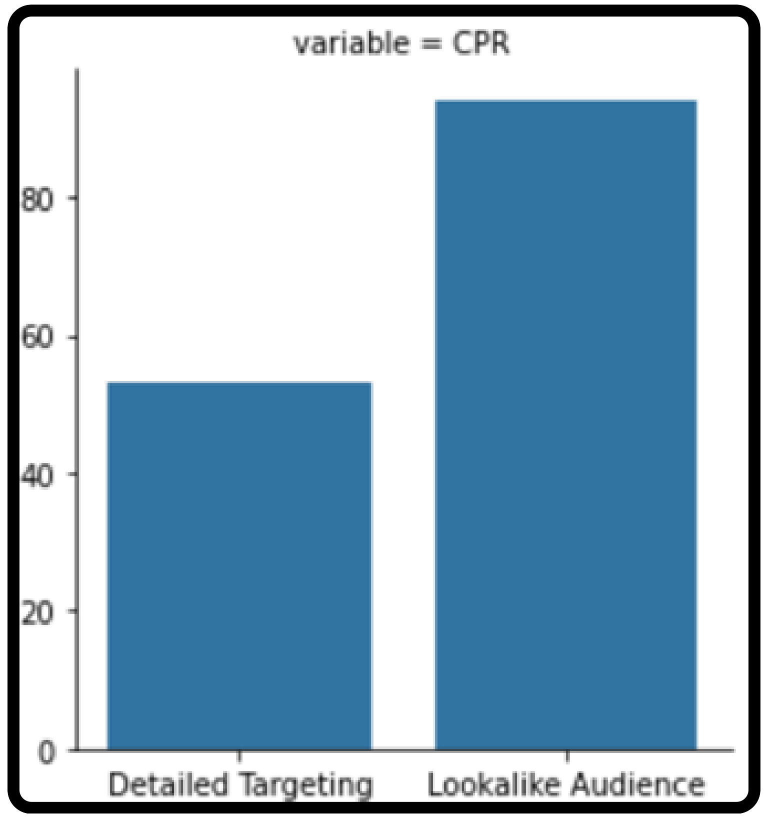
Reach

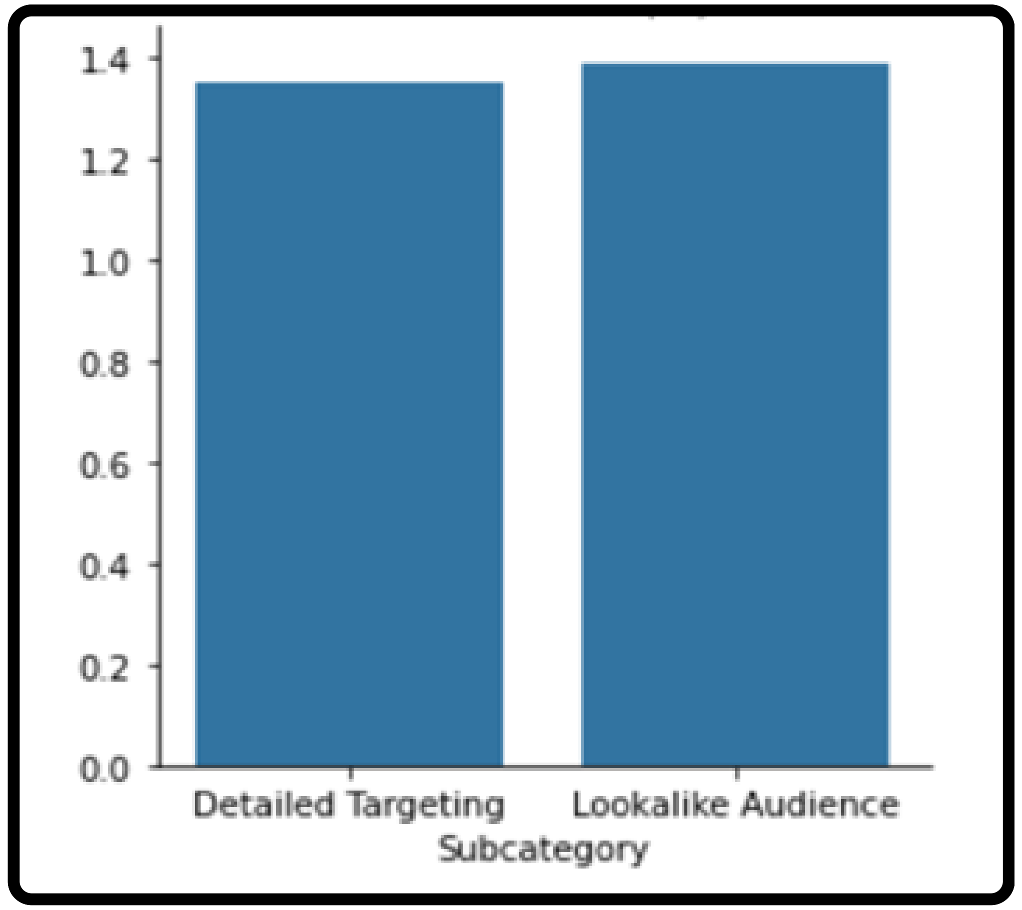


Impression

LOOKALIKE VS DETAILED TARGETING [RESULTS, CPR, AND CTR]







- **Results & CPR**
  - **Detailed** audiences achieve **more results** (conversions) than **Lookalike**.
  - **Detailed** audiences have a **lower cost** for conversions compared to **Lookalike**.

- **CTR (Click-Through Rate)**

No significant differences in **CTR** were observed; **both audiences** engage **equally** with the content.

LOOKALIKE VS DETAILED TARGETING [CONCLUSION]

- **Lookalike audiences**
  - They are **more extensive** and **suitable** for **brand awareness campaigns.**

- **Detailed audiences**
  - Despite being **smaller**, are more **likely to convert**, making them ideal for **conversion-focused campaigns.**

VARIOUS FUNNEL METRICS

Go back to Agenda Page

VARIOUS FUNNEL METRICS PLOTS CODING

TOP FUNNEL METRICS

```
In [89]: def top_funnel_metrics(df):
             fig,axs = plt.subplots(1, 3, figsize=(23,5))

             sns.barplot(x='Ad Set Name', y='Reach', data=df, ax=axs[0])
             axs[0].set_title('Reach')

             sns.barplot(x='Ad Set Name', y='Impressions', data=df, ax=axs[1])
             axs[1].set_title('Impressions')

             sns.barplot(x='Ad Set Name', y='CPM', data=df, ax=axs[2])
             axs[2].set_title('CPM')

             for ax in axs:
                 for label in ax.get_xticklabels():
                     label.set_rotation(90)

             plt.show()

         top_funnel_metrics(df_hot)
```

MIDDLE FUNNEL METRICS

```
In [90]: def mid_funnel_metrics(df):
             fig, axs = plt.subplots(1, 3, figsize=(23,5))

             sns.barplot(x='Ad Set Name', y='CTR (all)', data=df, ax=axs[0])
             axs[0].set_title('CTR (all)')

             sns.barplot(x='Ad Set Name', y='Landing Page Views', data=df, ax=axs[1])
             axs[1].set_title('Landing Page Views')

             sns.barplot(x='Ad Set Name', y='CPC', data=df, ax=axs[2])
             axs[2].set_title('CPC')

             for ax in axs:
                 for label in ax.get_xticklabels():
                     label.set_rotation(90)

             plt.show()

         mid_funnel_metrics(df_hot)
```

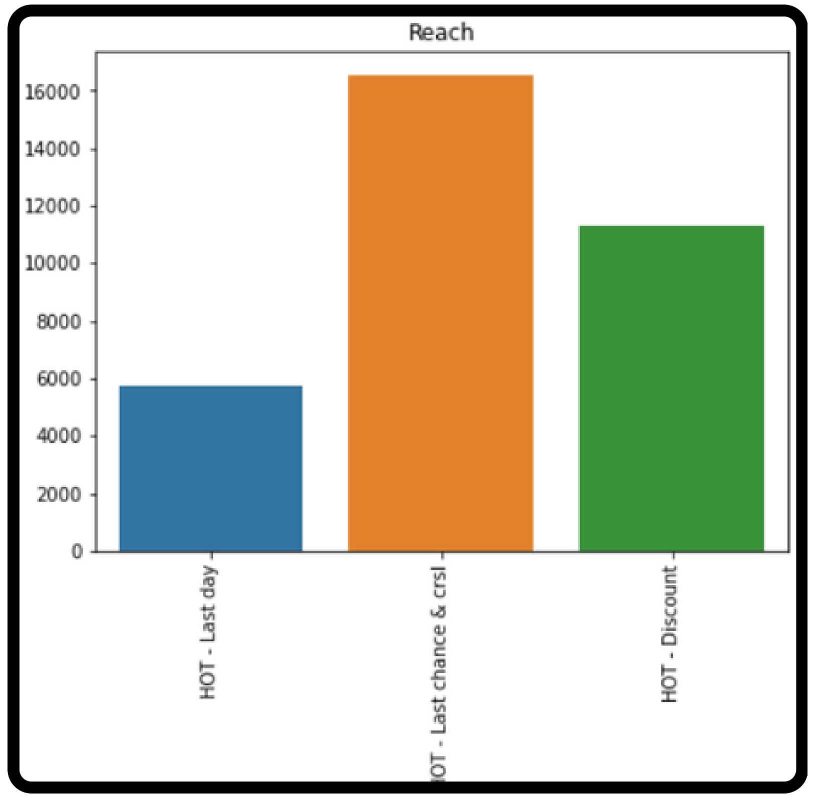
BOTTOM FUNNEL METRICS

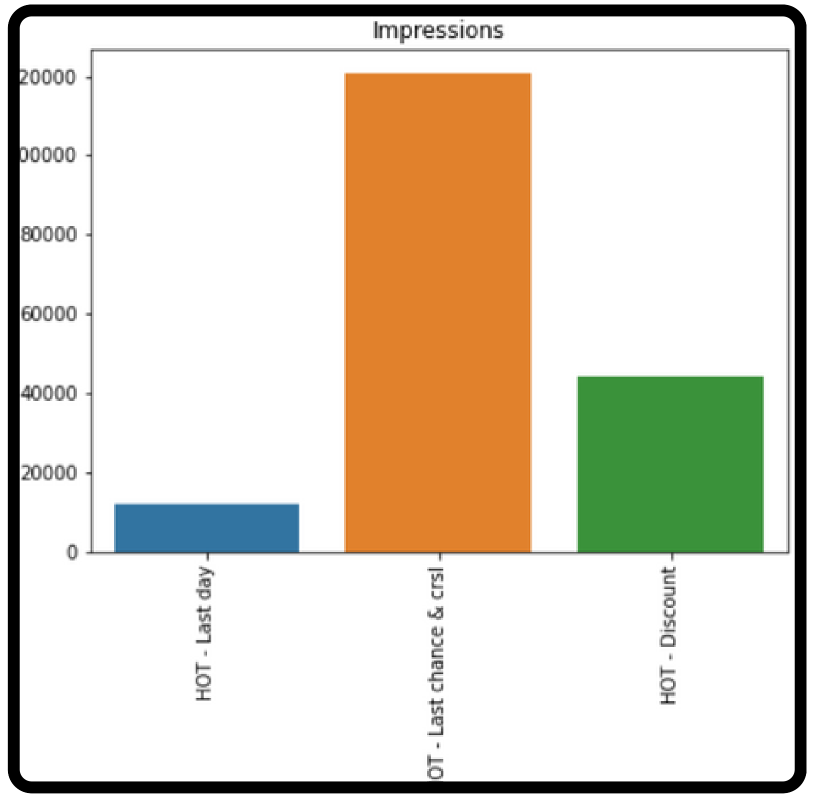
```
In [91]: def bottom_funnel_metrics(df):
             fig, axs = plt.subplots(1, 2, figsize=(23,5))

             sns.barplot(x='Ad Set Name', y='Results', data=df, ax=axs[0])
             axs[0].set_title('Results')

             sns.barplot(x='Ad Set Name', y='CPR', data=df, ax=axs[1])
             axs[1].set_title('CPR')

             for ax in axs:
                 for label in ax.get_xticklabels():
                     label.set_rotation(90)

             plt.show()

         bottom_funnel_metrics(df_hot)
```

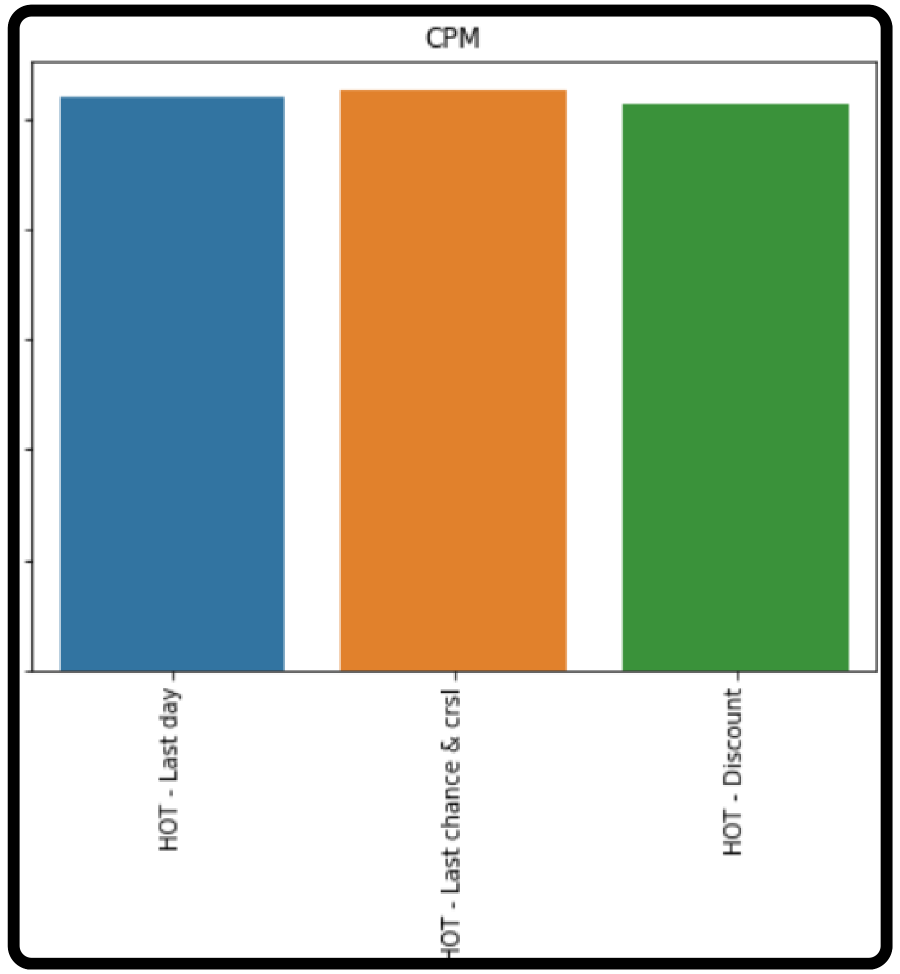
STATISTICS & CHARTS

Go back to Agenda Page

TOP FUNNEL METRICS ANALYSIS
[HOT AD SET]







- **<u>Reach & Impression</u>**

The ad set (**Last - Chance & Crsl**) outperforms others with **higher** <u>reach</u> and <u>impressions</u>, indicating a **wider user reach**.

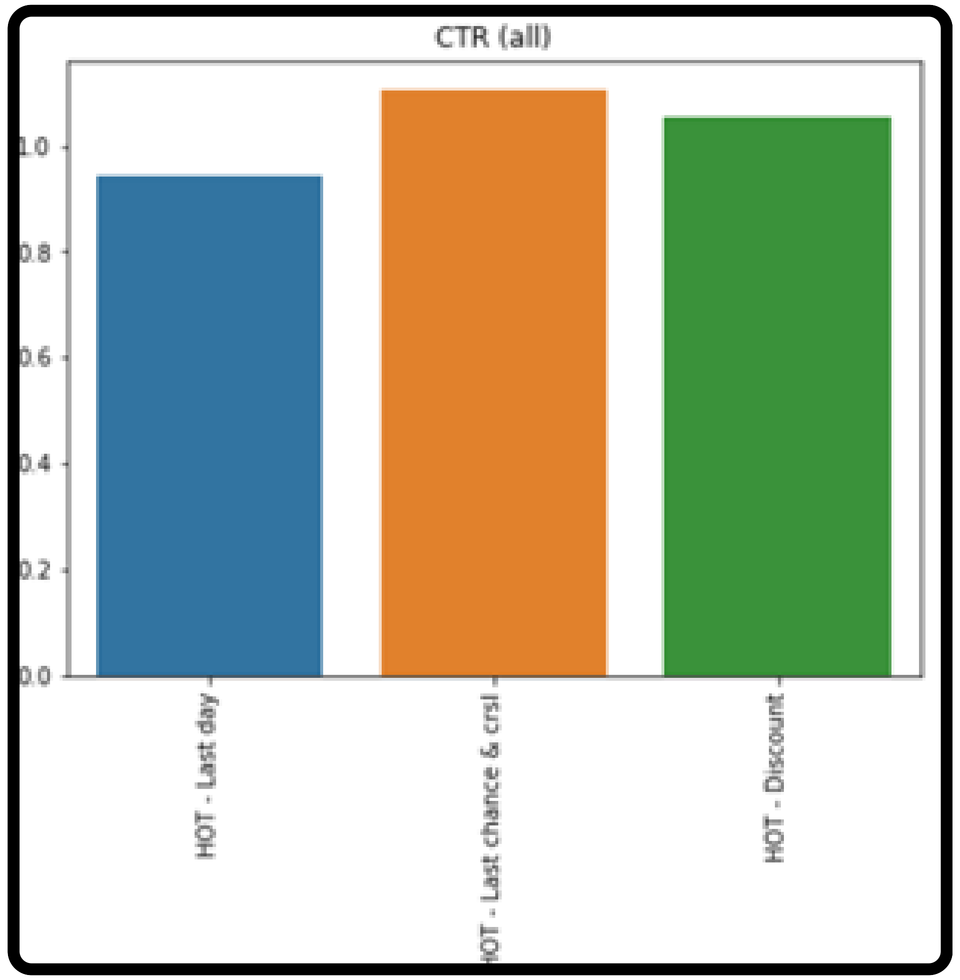
- **<u>CTR (Click-Through Rate)</u>**

All **three** ad sets exhibit **similar costs per 1000 impressions**, suggesting **no significant** cost differences among them.

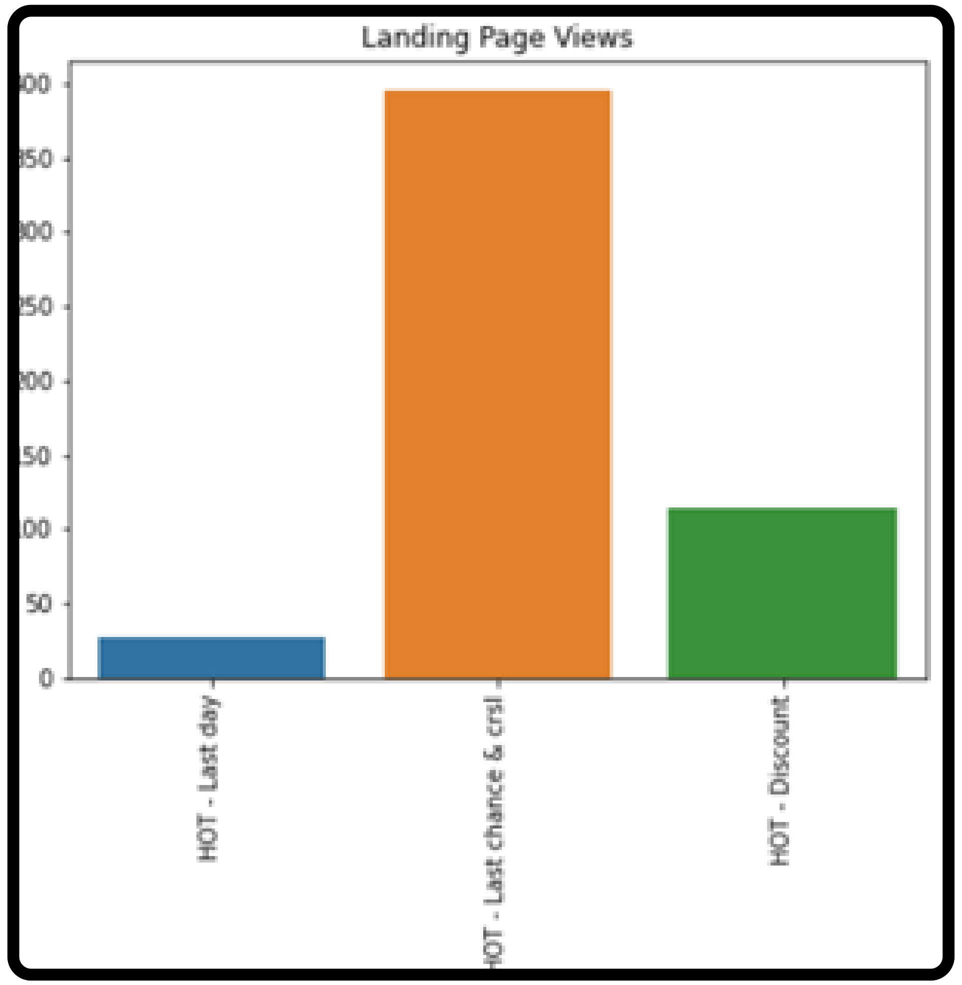
MIDDLE FUNNEL METRICS ANALYSIS [HOT AD SET]

- **Middle Funnel Metrices**

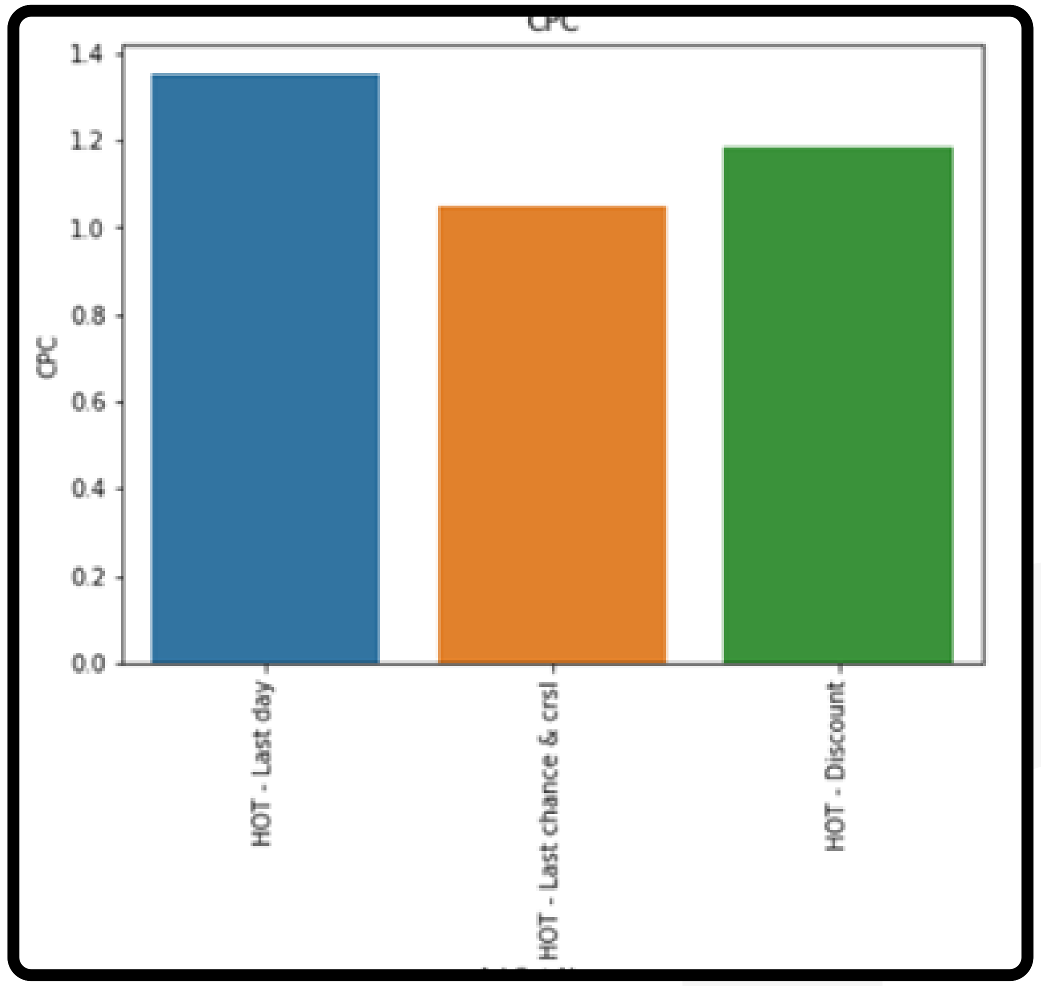
The ad set (**Last - Chance & Crsl**) shows **higher CTR** and **views** with **low CPC** which means **more engaging users**.



CTR



Landing Page Views

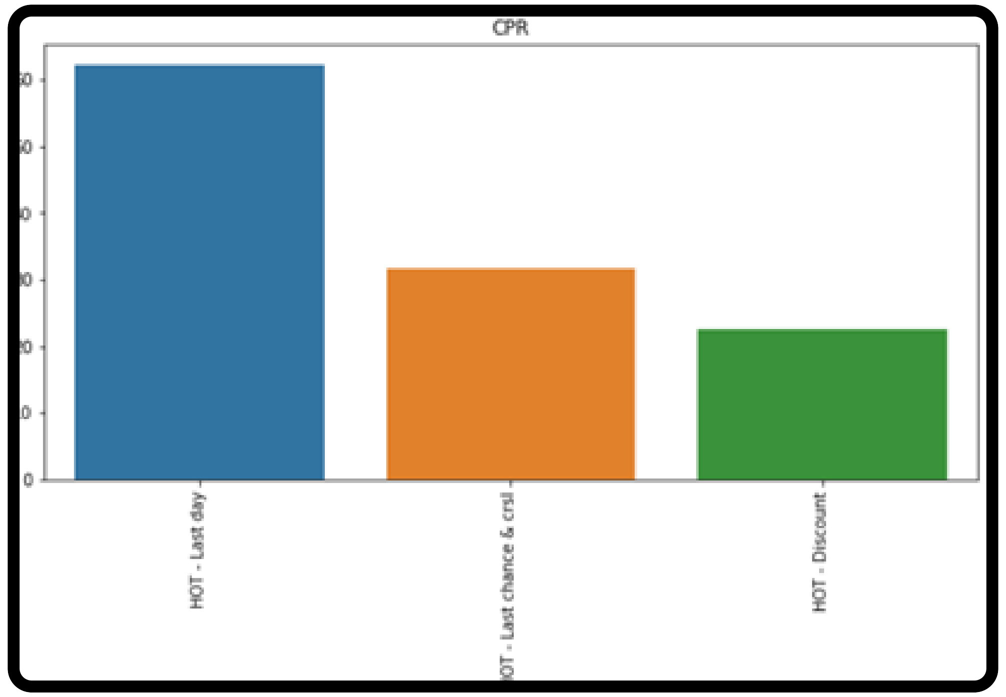


CPC

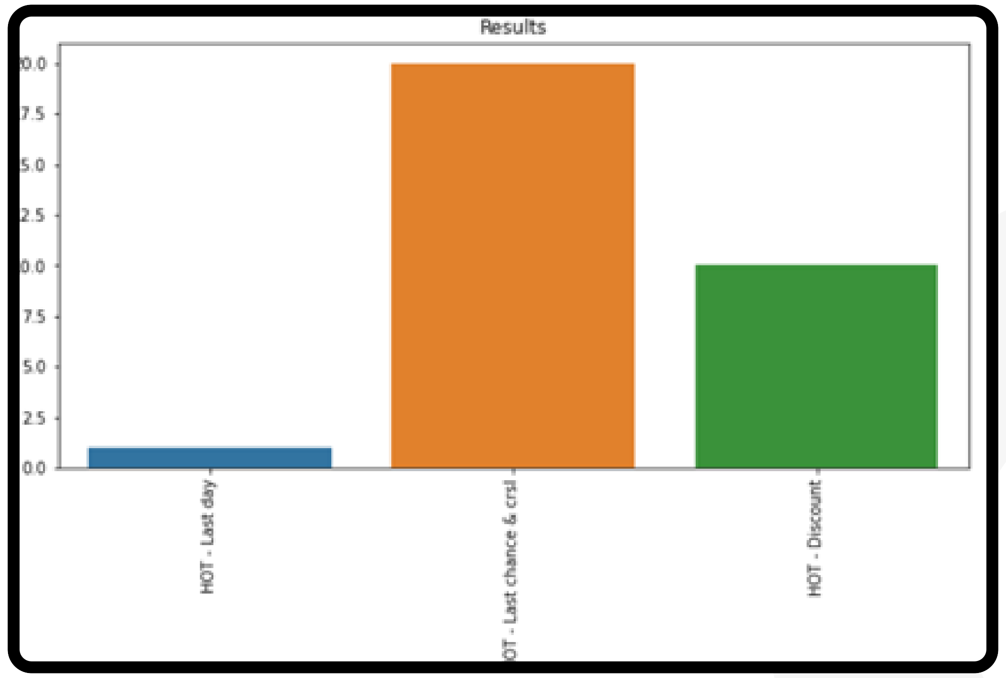
BOTTOM FUNNEL METRICS ANALYSIS [HOT AD SET]

CPR & Results

- **(Last - Chance & Crsl) and Discount Ad Sets:**
    - **(Last - Chance & Crsl)** achieves **higher results** but has a **higher CPR** (Cost Per Result) at **31** Euro.
    - **Discount ad set** is **more cost-effective** with a **CPR** of **22** Euro, making it **cheaper** for obtaining **results**.
- **Last Day Ad Set:**
    - It performs poorly with **fewer results** and the **highest CPR**, indicating it might not be as effective in achieving desired outcomes compared to other Hot ad sets.



CPC

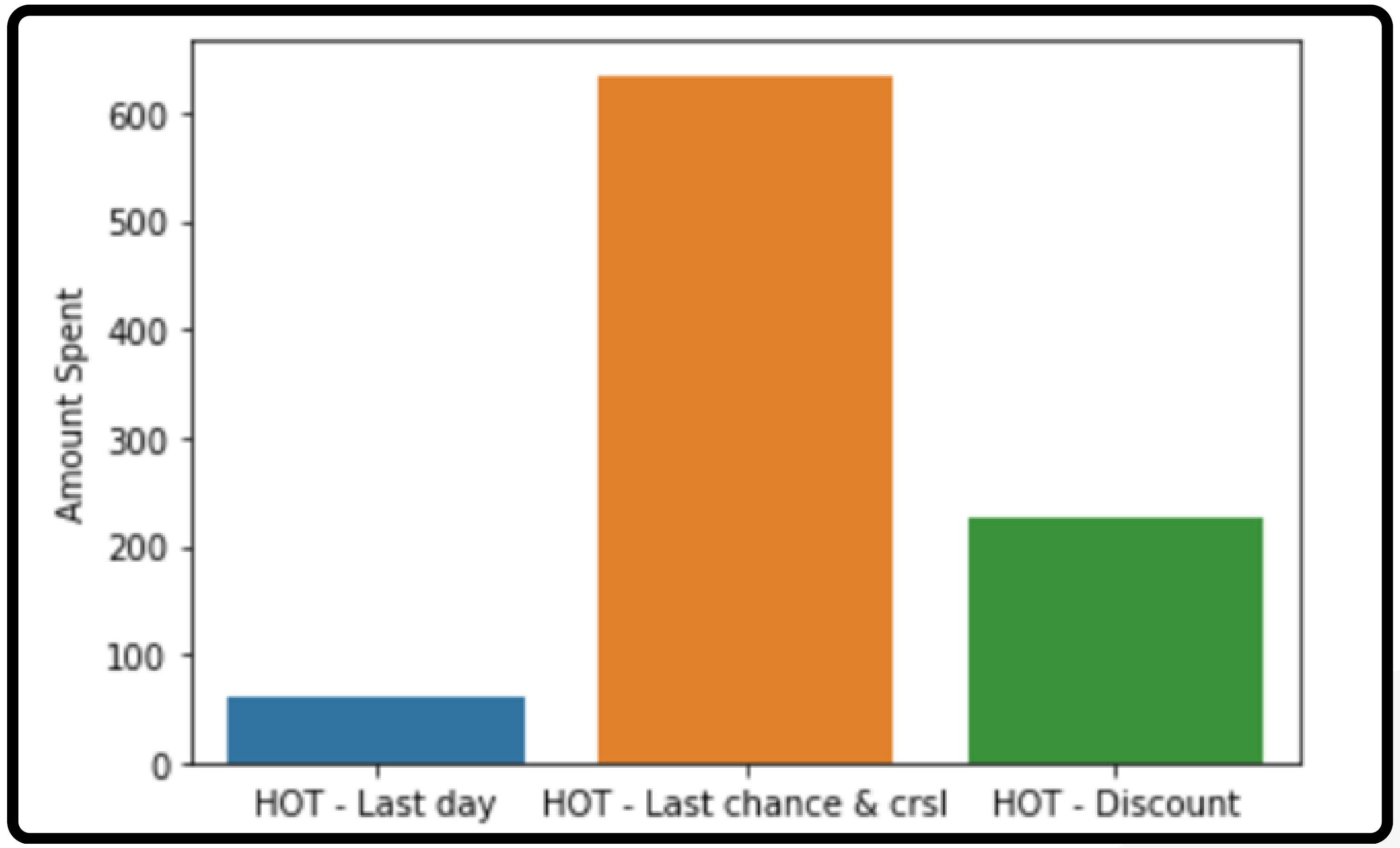


Results

BOTTOM FUNNEL METRICS ANALYSIS [HOT AD SET]

Ad Set Amount Spent

- The company spent around **600 euros** on Hot (**Last - Chance & Crsl**).
- (Last - Chance & Crsl) has a **higher** overall **cost** than Hot-Discount, which had an expenditure of around **210 euros**.
- Despite the **lower** spending, Hot-Discount is **more cost-effective**, with a **lower cost per result**.
- **Suggesting** it might be advisable to **allocate more budget** to this Hot-Discount Ad set for potentially better results at a lower cost per outcome.



THANK YOU

Go back to Agenda Page

PROJECT'S LINK:
Click to Explore The Code