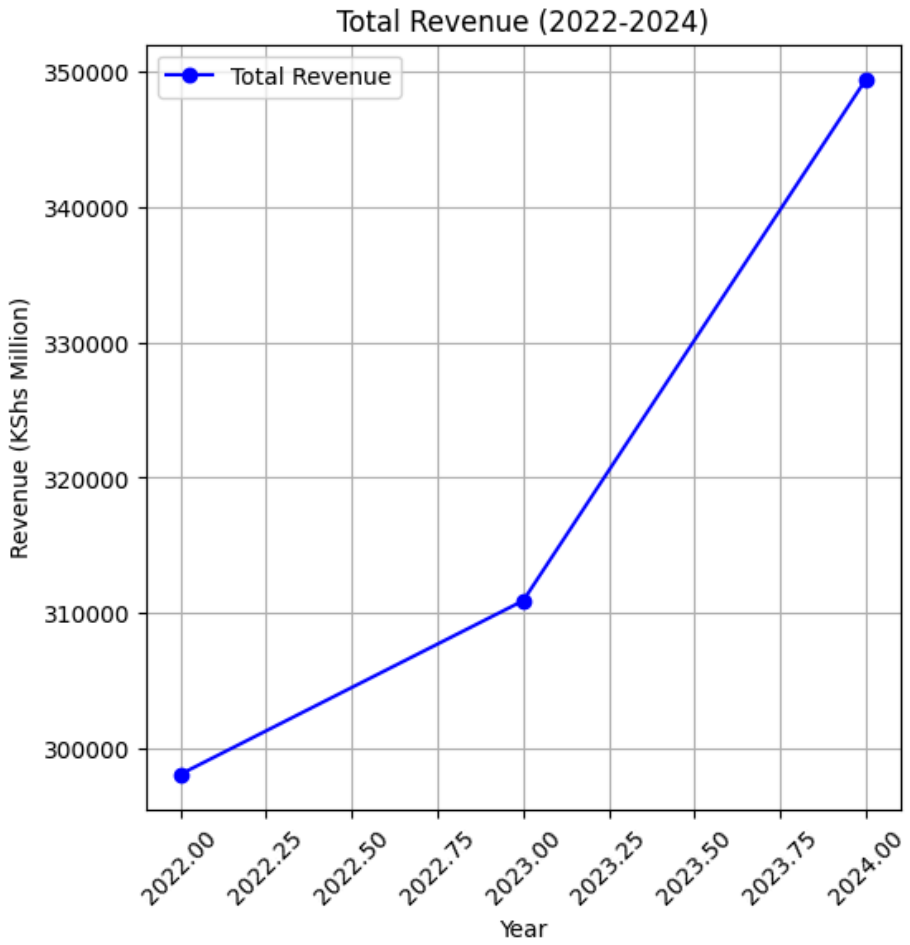
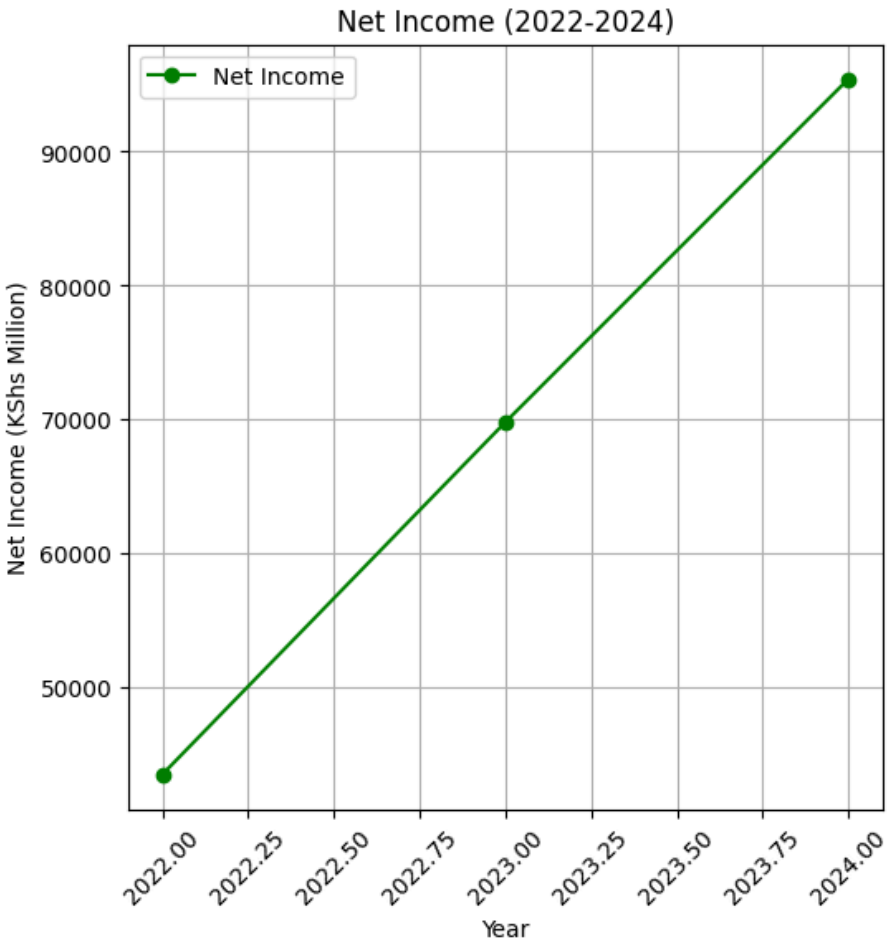
In [14]:
```
# Importing necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Data for Safaricom (2022-2024)
data = {
    "Year": [2022, 2023, 2024],
    "Total_Revenue": [298077.90, 310904.80, 349447.20],  # In KShs Million
    "Net_Income": [43495.27, 69795.18, 95412.03],        # In KShs Million
    "EBITDA": [153084.10, 163072.40, 185235.90],         # In KShs Million
    "Depreciation": [82948, 54865, 39933],               # In KShs Million
    "CapEx": [84000, 58000, 37000],                      # In KShs Million
    "Change_in_NWC": [3000, 3000, 9000]                  # In KShs Million
}

# Convert data to DataFrame
df = pd.DataFrame(data)



# Calculating Revenue Growth Rate
df['Growth_Rate'] = df['Total_Revenue'].pct_change() * 100

# Calculating EBITDA Margin
df['EBITDA_Margin'] = (df['EBITDA'] / df['Total_Revenue']) * 100

# Calculating Net Profit Margin
df['Net_Profit_Margin'] = (df['Net_Income'] / df['Total_Revenue']) * 100



# Displaying the DataFrame
print("Financial Performance Data:")
print(df)

# Calculate Summary Statistics
summary_stats = df[['Total_Revenue', 'Net_Income', 'EBITDA', 'Depreciation', 'Growth_Rate', 'EBITDA_Margin', 'Net_Profit_Margi
print("\nSummary Statistics:")
```

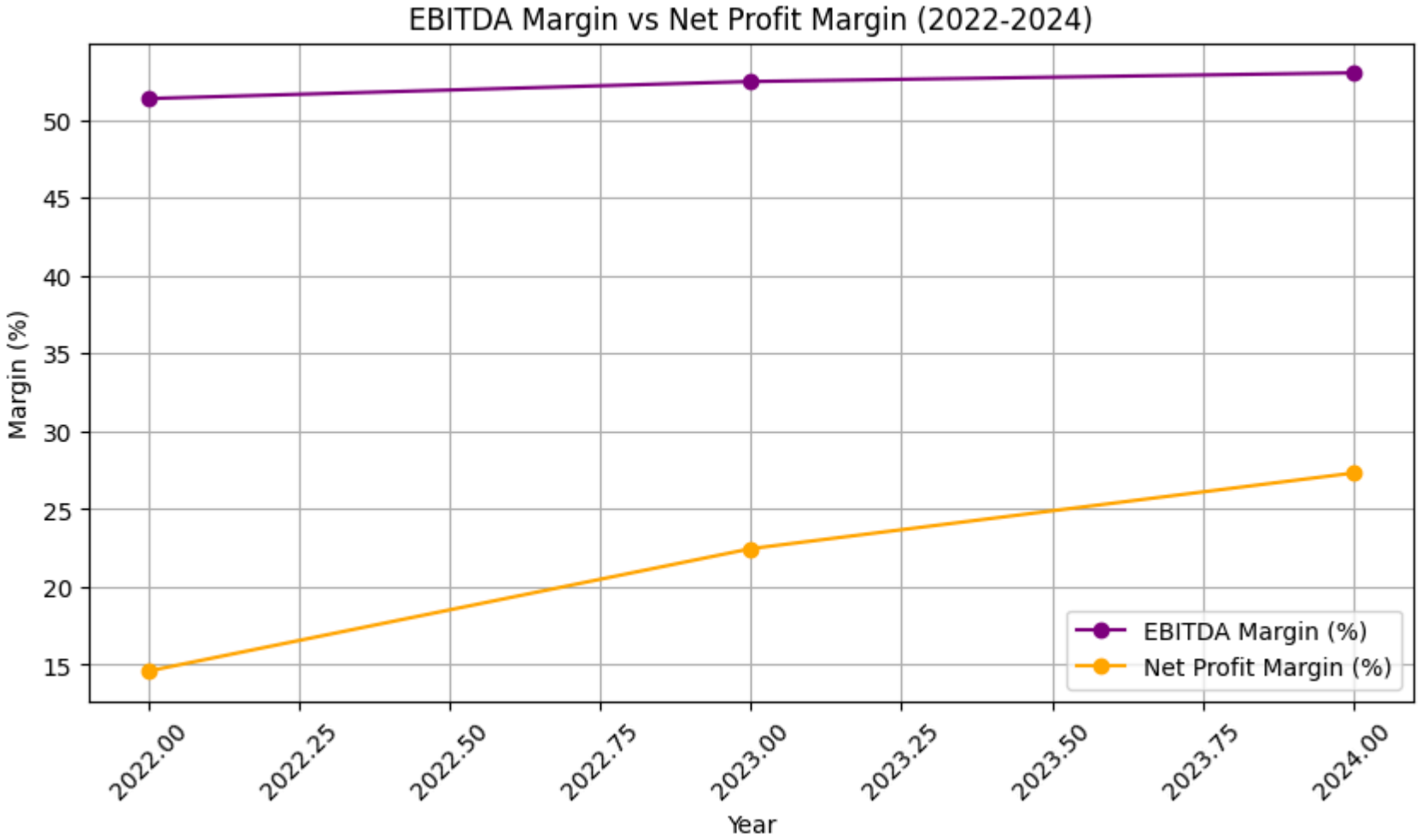
```
print(summary_stats)



# Plotting Revenue, Net Income
plt.figure(figsize=(20, 6))

# Plotting Total Revenue
plt.subplot(1, 3, 1)
plt.plot(df['Year'], df['Total_Revenue'], marker='o', color='b', label='Total Revenue')
plt.title('Total Revenue (2022-2024)')
plt.xlabel('Year')
plt.xticks(rotation=45)  # Rotate labels by 45 degrees
plt.ylabel('Revenue (KShs Million)')
plt.grid(True)
plt.legend()



# Plotting Net Income
plt.subplot(1, 3, 2)
plt.plot(df['Year'], df['Net_Income'], marker='o', color='g', label='Net Income')
plt.title('Net Income (2022-2024)')
plt.xlabel('Year')
plt.xticks(rotation=45)  # Rotate labels by 45 degrees
plt.ylabel('Net Income (KShs Million)')
plt.grid(True)
plt.legend()


# Plotting Margins
plt.figure(figsize=(10, 5))
plt.plot(df['Year'], df['EBITDA_Margin'], marker='o', color='purple', label='EBITDA Margin (%)')
plt.plot(df['Year'], df['Net_Profit_Margin'], marker='o', color='orange', label='Net Profit Margin (%)')
plt.title('EBITDA Margin vs Net Profit Margin (2022-2024)')
plt.xlabel('Year')
plt.xticks(rotation=45)  # Rotate labels by 45 degrees
plt.ylabel('Margin (%)')
plt.grid(True)
```

```
plt.legend()
plt.show()
```

```
Financial Performance Data:
   Year  Total_Revenue  Net_Income    EBITDA  Depreciation  CapEx  \
0  2022       298077.9    43495.27  153084.1         82948  84000
1  2023       310904.8    69795.18  163072.4         54865  58000
2  2024       349447.2    95412.03  185235.9         39933  37000

   Change_in_NWC  Growth_Rate  EBITDA_Margin  Net_Profit_Margin
0           3000          NaN      51.357078          14.591914
1           3000     4.303204      52.450911          22.449052
2           9000    12.396849      53.008266          27.303704

Summary Statistics:
       Total_Revenue    Net_Income         EBITDA  Depreciation  Growth_Rate  \
count       3.000000      3.000000       3.000000      3.000000     2.000000
mean   319476.633333  69567.493333  167130.800000  59248.666667     8.350027
std     26735.902652  25959.128898   16455.622709  21839.985264     5.723072
min    298077.900000  43495.270000  153084.100000  39933.000000     4.303204
25%    304491.350000  56645.225000  158078.250000  47399.000000     6.326615
50%    310904.800000  69795.180000  163072.400000  54865.000000     8.350027
75%    330176.000000  82603.605000  174154.150000  68906.500000    10.373438
max    349447.200000  95412.030000  185235.900000  82948.000000    12.396849

       EBITDA_Margin  Net_Profit_Margin
count       3.000000           3.000000
mean       52.272085          21.448223
std         0.839994           6.414721
min        51.357078          14.591914
25%        51.903995          18.520483
50%        52.450911          22.449052
75%        52.729588          24.876378
max        53.008266          27.303704
```


Total Revenue (2022-2024)
Total Revenue
Revenue (KShs Million)
350000
340000
330000
320000
310000
300000
2022.00
2022.25
2022.50
2022.75
2023.00
2023.25
2023.50
2023.75
2024.00
Year

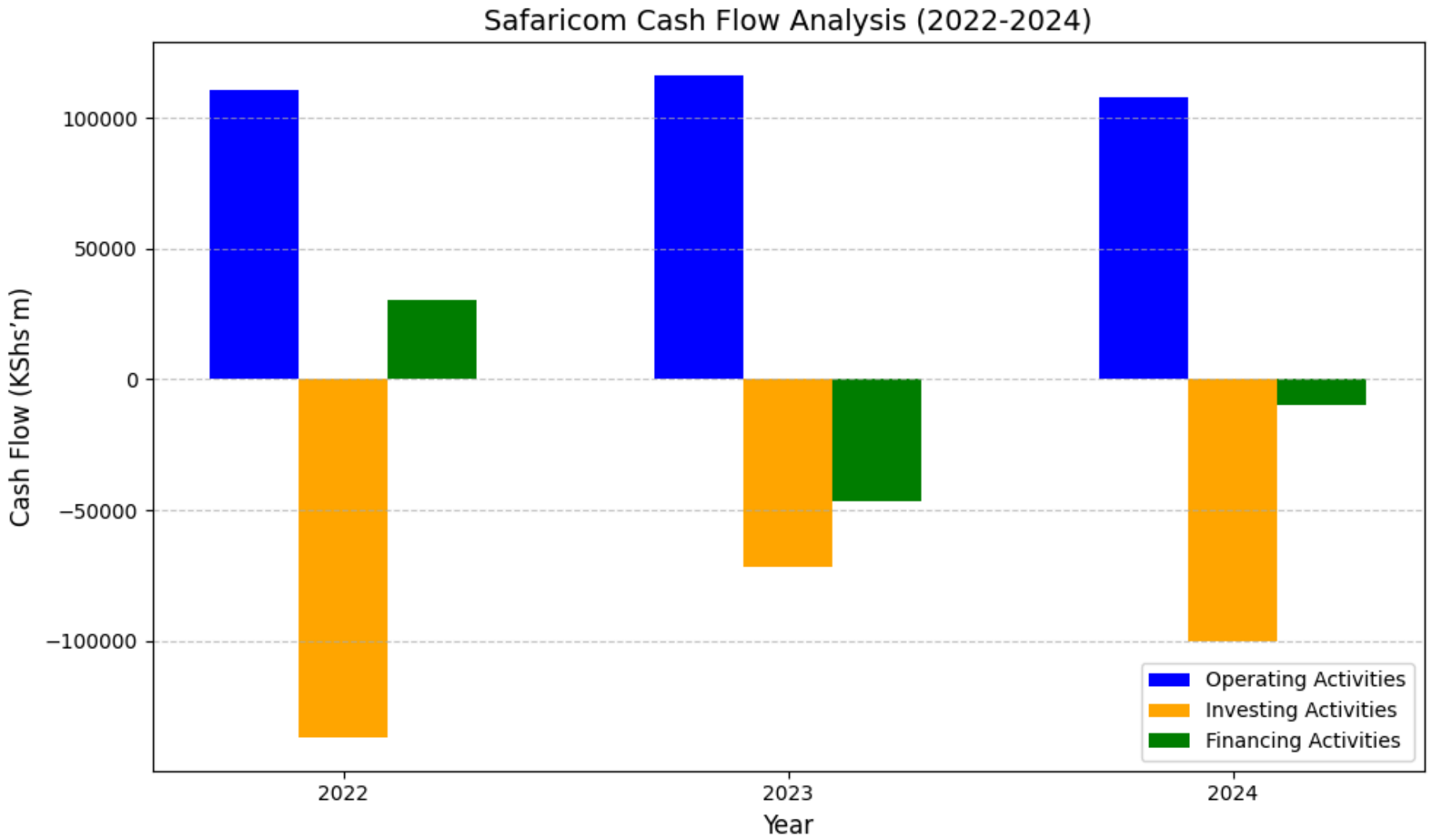

Net Income (2022-2024)
Net Income
Net Income (KShs Million)
90000
80000
70000
60000
50000
2022.00
2022.25
2022.50
2022.75
2023.00
2023.25
2023.50
2023.75
2024.00
Year



CASH FLOW ANALYSIS - SAFARICOM

In [17]:
```
# Extracting data Manually  from the cash flow statements
data = {
    'Year': [2022, 2023, 2024],
    'Operating Activities (KShs'm)': [110700.5, 116151.1, 107923.6],
    'Investing Activities (KShs'm)': [-136944.8, -71875.6, -100218.2],
    'Financing Activities (KShs'm)': [30382.2, -46864.4, -9696.5]
```

```
}

# Creating DataFrame
df = pd.DataFrame(data)

# Setting up the bar chart
fig, ax = plt.subplots(figsize=(10, 6))

# Defining bar width and position of each bar group
bar_width = 0.2
x = df['Year']

# Creating bars for each category
ax.bar(x - bar_width, df['Operating Activities (KShs’m)'], width=bar_width, label='Operating Activities', color='blue')
ax.bar(x, df['Investing Activities (KShs’m)'], width=bar_width, label='Investing Activities', color='orange')
ax.bar(x + bar_width, df['Financing Activities (KShs’m)'], width=bar_width, label='Financing Activities', color='green')

# Customizing the plot
ax.set_title('Safaricom Cash Flow Analysis (2022-2024)', fontsize=14)
ax.set_xlabel('Year', fontsize=12)
ax.set_ylabel('Cash Flow (KShs’m)', fontsize=12)
ax.legend()

# Showing the plot
plt.xticks(df['Year'])
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```


Safaricom Cash Flow Analysis (2022-2024)
Cash Flow (KShs'm)
100000
50000
0
−50000
−100000
2022
2023
2024
Year
Operating Activities
Investing Activities
Financing Activities