

REPORT

ON

NFT SALES ANALYTICS DASHBOARD

## TABLE OF CONTENTS

1. Introduction.....	3
1.1.Purpose.....	4
1.2.Objectives.....	5
1.3.Scope.....	6
2. Literature Survey.....	7
3. Problem Definition.....	8
4. Proposed System.....	9
4.1. Features and Functionality.....	10
5. Software Requirements .....	11
6. Implementation.....	12
7. Results.....	18
8. Conclusion.....	23
9. Future Scope.....	24

References

## **1. INTRODUCTION**

A non-fungible token (NFT) is a unique digital identifier that cannot be copied, substituted, or subdivided, that is recorded in a blockchain, and that is used to certify authenticity and ownership. The ownership of an NFT is recorded in the blockchain and can be transferred by the owner, allowing NFTs to be sold and traded. NFTs can be created by anybody, and require few or no coding skills to create. NFTs typically contain references to digital files such as photos, videos, and audio. Because NFTs are uniquely identifiable assets, they differ from cryptocurrencies, which are fungible.

Proponents of NFTs claim that NFTs provide a public certificate of authenticity or proof of ownership, but the legal rights conveyed by an NFT can be uncertain. The ownership of an NFT as defined by the blockchain has no inherent legal meaning and does not necessarily grant copyright, intellectual property rights, or other legal rights over its associated digital file. An NFT does not restrict the sharing or copying of its associated digital file and does not prevent the creation of NFTs that reference identical files.

The NFT market grew dramatically from 2020–2021: the trading of NFTs in 2021 increased to more than \$17 billion, up by 21,000% over 2020's total of \$82 million. NFTs have been used as speculative investments, and they have drawn increasing criticism for the energy cost and carbon footprint associated with validating blockchain transactions as well as their frequent use in art scams. The NFT market has also been compared to an economic bubble or a Ponzi scheme. By May 2022, the NFT market was seen as beginning to collapse.

Research on NFTs is still limited, and focuses mostly on technical aspects, such as copyright regulations; components, protocols, standards, and desired properties; new blockchain-based protocols to trace physical goods and the implications that NFTs have on the art world, in particular as they allow to share secondary sale royalties with the artist. Empirical studies aiming at characterizing properties of the market have focused on a limited number of NFT collections, such as CryptoKitties, Cryptopunks, and Axie, or on a single NFT market, such as Decentraland or super rare. These analyses revealed that the digital abundance of NFTs in digital games has led to a substantial decrease in their value, and that, even if NFT prices are driven by the prices of cryptocurrencies, the NFT market could be prone to speculation. Further, it was shown that NFTs valued by experts are more successful, and that, based on 16 000 NFTs sold on the super rare market, the structure of the NFT co-ownership network is highly centralized and small-world-like

## **1.1. PURPOSE**

There are certain attributes of NFTs that bring certain benefits.

- Non-fungible tokens are limited in their supply. It is the scarcity of their supply that brings it value. While there's a possibility that NFT developers can have an unlimited supply of non-fungible tokens, they ensure that the supply is limited, so there is an ongoing interest.
- The non-fungible token is indivisible. You either buy the entire token or don't buy it at all. While this might directly contradict the possibilities of partial ownership, it is designed to facilitate complete ownership of certain valuables and unique assets.
- Needless to say, an NFT is unique and they have a strong repository of information. These pieces of information make it unique and the information is fully secured and genuine.

## **1.2. OBJECTIVES**

- To create a dashboard that can visualize NFT Data.
- To clean and replace missing values.
- To use preprocessing methods available in pandas.
- To grab insights from the data that can help buyers as well as Sellers.
- To create a user-friendly user interface.
- To analyze data using IBM Cognos Analytics.

### 1.3. SCOPE

NFTs are not currencies. They are digital tokens or an address generated on a blockchain, which is completely unique and cannot be copied. No two NFTs are the same. This makes NFT a universally accepted and verifiable digital asset. NFTs are a sort of digital ownership of an asset recorded in the blockchain. These assets can be digital or physical in nature. NFTs are similar in this regard to fungible goods such as cryptocurrencies. However, they differ in that fungible goods can be exchanged for goods of the same kind. By contrast, non-fungible items cannot be exchanged for a similar good because they are unique. Prominent examples of non-fungible tokens (NFT's), such as the artist Beeple selling his digital art 'Everyday' for \$69 million (Christie's, 2021) and Twitter CEO Jack Dorsey auctioning off his first tweet for \$2.9 million (The Verge), shows NTFs have received mainstream attention.

Public attention towards NFTs has exploded in 2021 when their market experienced record sales, but little is known about the overall structure and evolution of its market. The scope of NFT sales analytics is huge. We've seen the rise of several different types of NFTs, including collectibles, games, art, and even food. There are many ways to use NFTs, and each method has its own unique set of pros and cons.

Collectible NFTs are the most popular type of NFT. These are items that are designed to be collected over time. A good example of this would be a piece of artwork that you buy and display on your wall. You could then go back later and purchase additional pieces of artwork to add to your collection. Another popular way to use NFTs is to play games with them. Games are fun, interactive experiences where users earn points based on their actions. Users can then trade these points for prizes. Popular examples of games include CryptoKitties, Etheremon, and Augmented Reality (AR) games.

The project focuses on analyzing the data of NFTs and creating a dashboard with the most recent sales of NFT's. The Dashboard contains different visualizations showing the data related to NFT collections, sales, etc.

## **2. LITERATURE SURVEY**

**1. S. Bouraga, "On the Popularity of Non-Fungible Tokens: Preliminary Results," 2021 3rd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), 2021, pp. 49-50, doi: 10.1109/BRAINS52497.2021.9569792.**

The authors state that Blockchain technology supports digital assets, which can take the form of cryptocurrencies and tokens. Tokens are usually created on top of the blockchain platform, using smart contracts. Two main categories of tokens exist: Fungible Tokens and Non-Fungible Tokens (NFTs). Here, they focused on NFTs and proposed a correlation analysis between various NFTs' characteristics and the popularity of the NFTs. The results can have practical implications for both designers and users.

**2. S. Bradić, D. Delija, G. Sirovatka and M. Žagar, "Creating own NFT token using ERC721 standard and solidity programming language," 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), 2022, pp. 1053-1056, doi: 10.23919/MIPRO55190.2022.9803593.**

The paper analyzes which industries blockchain can be an advantage and how it can improve and make the user experience better and more secure. Blockchain also carries with it potential "threats" related to the way data is stored, so the most common possible ways of damaging the blockchain are analyzed. In the practical part of the paper, the article describes how the reader can independently make an NFT token based on the ERC721 standard Ethereum blockchain and how to set it on the blockchain and distribute, and potentially one can make money on it, which is another case of using the blockchain.

**3. A. Park, J. Kietzmann, L. Pitt and A. Dabirian, "The Evolution of Non Fungible Tokens: Complexity and Novelty of NFT Use-Cases," in IT Professional, vol. 24, no. 1, pp. 9-14, 1 Jan.-Feb. 2022, doi: 10.1109/MITP.2021.3136055.**

In this paper, they have examined the emergence of NFTs, from their technical origins, the introduction of blockchain technologies and the first token-based collectibles that led to modern day NFT products. They have categorized the current use cases for NFTs, introduce their potential future applications, and highlight the challenges managers face in incorporating them into their existing workflows. By presenting our NFT adoption framework, we offer managers strategies for evaluating the risks and benefits of NFTs.

### **3. PROBLEM DEFINITION**

The NFT market is booming now. According to NFTGO statistics, at least one NFT project has been launched on the chain every day since May 2021. In the world of blockchain, whether as an investor or a beginner, you can rely on data to stay away from “hearsay” and decide which NFT to buy by yourself. Analytics dashboards and reports can be created to know more about NFT. By using analytics tools to identify patterns and discrepancies, traders, builders, and collectors can gain a competitive edge in a brand-new market.

The problem definition for the NFT sales analytics dashboard is to provide a simple way for users to understand their data. Users should have access to information about how they are performing compared to the industry average. Data should be presented in a format that makes sense to them and provides them with actionable insights. The problem we want to solve is how to analyze the data generated by our NFT. We have a python application where users can visualize their NFTs. Our analytics dashboard will help users gain insight into their business performance, and allow them to make informed decisions about their operations.



## **4. PROPOSED SYSTEM**

Companies making nft sales clearly benefit from having good analytics software in place. These systems make it easy for them to manage their nft sales operations efficiently and effectively. Nft sales channels are readily available thanks to modern technology allowing consumers new ways to purchase products from companies. Ultimately, better analytics lead to a better world.

We propose a solution that uses a combination of charts, graphs, and tables to present our data in a clear and concise manner. We use line charts to show the distribution of each metric across different periods and to compare these metrics over time.

Dashboards are a great way to visualize data and present information in a clear and concise manner. In this case, we are proposing a dashboard design for the proposed system for NFT sales analysis. The proposed design for the Sales Analytics dashboard consists of two major components: (a) the front-end interface that contains visualization and (b) the back-end service layer that processes the data and returns the results. The front-end interface is responsible for receiving the data from the users and displaying them in a way that makes sense for the end user. The back-end service layer is responsible for processing the data received from the front-end interface and returning the results.

The first step in designing any dashboard is collecting the right data. Here, we have collected data from Kaggle. Once the data is collected, it would be time to visualize the data. Here, we are using graphs to display the data.

#### **4.1. FEATURE AND FUNCTIONALITY**

- NFT sellers use these insights to optimize the performance of their in-store campaigns by making necessary store changes such as product placements, stock levels, and marketing strategies.
- Consumers use NFT sales analytics when making purchases from retailers- whether online or in person
- Processed NFT data helps both consumers and businesses understand their customer bases and make informed decisions about what products they should buy next.
- Analytics also help retailers plan new NFT sales campaigns that meet or exceed the expectations of consumers.

## **5. SOFTWARE REQUIREMENTS**

### **Python:**

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured, object-oriented, and functional programming

### **Tkinter:**

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

### **Matplotlib:**

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

### **Pandas:**

Pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labeled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real-world data analysis in Python.

### **VSCode:**

Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft with the Electron Framework, for Windows, Linux, and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

## 6. IMPLEMENTATION

### Step 1 : Visualization

```
[ ] from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive

[ ] import numpy as np
import pandas as pd

[ ] df=pd.read_csv("/content/drive/MyDrive/NFT_Sales.csv")
df.head()
```

	Date	Sales_USD_cumsum	Number_of_Sales_cumsum	Active_Market_Wallets_cumsum	Primary_Sales_cumsum	Secondary_Sales_cumsum	AverageUSD_cum	Sales_USD	Number_of_Sales	Active_Market
0	2017-06-22	0.00	0	0.0	0	NaN	NaN	NaN	NaN	
1	2017-06-23	1020.30	19	8.0	0	19.0	53.70	1020.30	19.0	
2	2017-06-24	2261.14	40	21.0	0	21.0	56.53	1240.84	21.0	
3	2017-06-25	2778.69	53	28.0	0	13.0	52.43	517.55	13.0	
4	2017-06-26	3203.32	67	34.0	0	14.0	47.81	424.63	14.0	

```
[ ] df.shape
(1606, 11)

[ ] df.describe()
```

	Sales_USD_cumsum	Number_of_Sales_cumsum	Active_Market_Wallets_cumsum	Primary_Sales_cumsum	Secondary_Sales_cumsum	AverageUSD_cum	Sales_USD	Number_of_Sales	Active_Market
count	1.606000e+03	1.606000e+03	1598.000000	1.606000e+03	1605.000000	1605.000000	1.605000e+03	1605.000000	15
mean	5.914905e+08	3.397818e+06	155847.829161	2.218615e+06	2749.373832	98.241209	6.372813e+06	6868.409346	5
std	1.720005e+09	2.329504e+06	136083.110285	1.432603e+06	5597.500898	179.880778	2.486102e+07	12610.348160	13
min	0.000000e+00	0.000000e+00	0.000000	0.000000e+00	-1024.000000	7.140000	-6.196720e+06	-2195.000000	-5
25%	5.129669e+07	1.451181e+06	81998.250000	9.735630e+05	662.000000	27.110000	4.605234e+04	2274.000000	1
50%	8.691480e+07	3.203392e+06	121914.000000	2.296888e+06	1025.000000	32.050000	8.218757e+04	3681.000000	1
75%	1.328586e+08	4.938128e+06	180948.250000	3.150981e+06	1846.000000	71.360000	3.528744e+05	5844.000000	2
max	1.023456e+10	1.102278e+07	803665.000000	6.609013e+06	69685.000000	928.490000	3.886364e+08	161732.000000	301

```
[ ] df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1606 entries, 0 to 1605
Data columns (total 11 columns):
#   Column                                Non-Null Count  Dtype
---  ---                                ---
0   Date                                  1606 non-null   object
1   Sales_USD_cumsum                     1606 non-null   float64
2   Number_of_Sales_cumsum               1606 non-null   int64
3   Active_Market_Wallets_cumsum         1598 non-null   float64
4   Primary_Sales_cumsum                 1606 non-null   int64
5   Secondary_Sales_cumsum                1605 non-null   float64
6   AverageUSD_cum                       1605 non-null   float64
7   Sales_USD                           1605 non-null   float64
8   Number_of_Sales                      1605 non-null   float64
9   Active_Market_Wallets                 1597 non-null   float64
10  Primary_Sales                         1605 non-null   float64
dtypes: float64(8), int64(2), object(1)
memory usage: 138.1+ KB
```

```
[ ] df.isnull().sum()

Date      0
Sales_USD_cumsum      0
Number_of_Sales_cumsum      0
Active_Market_Wallets_cumsum      8
Primary_Sales_cumsum      0
Secondary_Sales_cumsum      1
AverageUSD_cum      1
Sales_USD      1
Number_of_Sales      1
Active_Market_Wallets      9
Primary_Sales      1
dtype: int64

[ ] df['Active_Market_Wallets_cumsum'] = df['Active_Market_Wallets_cumsum'].fillna(df['Active_Market_Wallets_cumsum'].mean())
df['Secondary_Sales_cumsum'] = df['Secondary_Sales_cumsum'].fillna(df['Secondary_Sales_cumsum'].mean())
df['AverageUSD_cum'] = df['AverageUSD_cum'].fillna(df['AverageUSD_cum'].mean())
df['Sales_USD'] = df['Sales_USD'].fillna(df['Sales_USD'].mean())
df['Number_of_Sales'] = df['Number_of_Sales'].fillna(df['Number_of_Sales'].mean())
df['Active_Market_Wallets'] = df['Active_Market_Wallets'].fillna(df['Active_Market_Wallets'].mean())
df['Primary_Sales'] = df['Primary_Sales'].fillna(df['Primary_Sales'].mean())
```

```
df.isnull().sum()

Date      0
Sales_USD_cumsum      0
Number_of_Sales_cumsum      0
Active_Market_Wallets_cumsum      0
Primary_Sales_cumsum      0
Secondary_Sales_cumsum      0
AverageUSD_cum      1
Sales_USD      0
Number_of_Sales      0
Active_Market_Wallets      0
Primary_Sales      0
AverageUSD_cum      0
dtype: int64

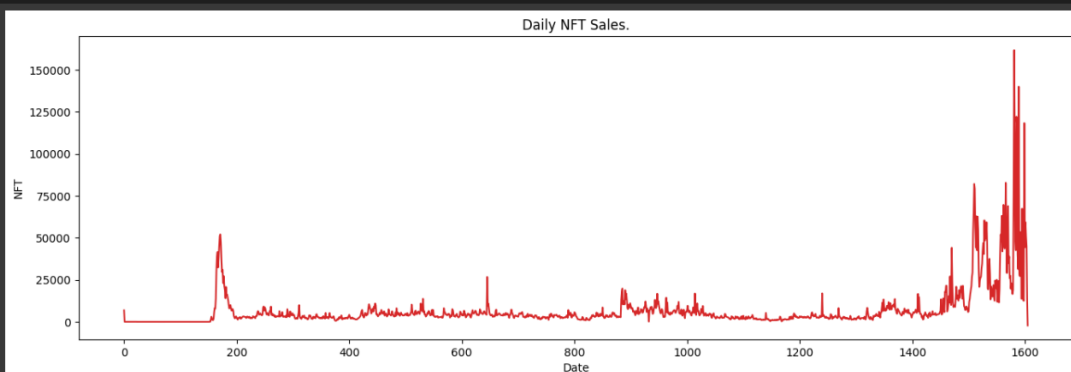
[ ] df

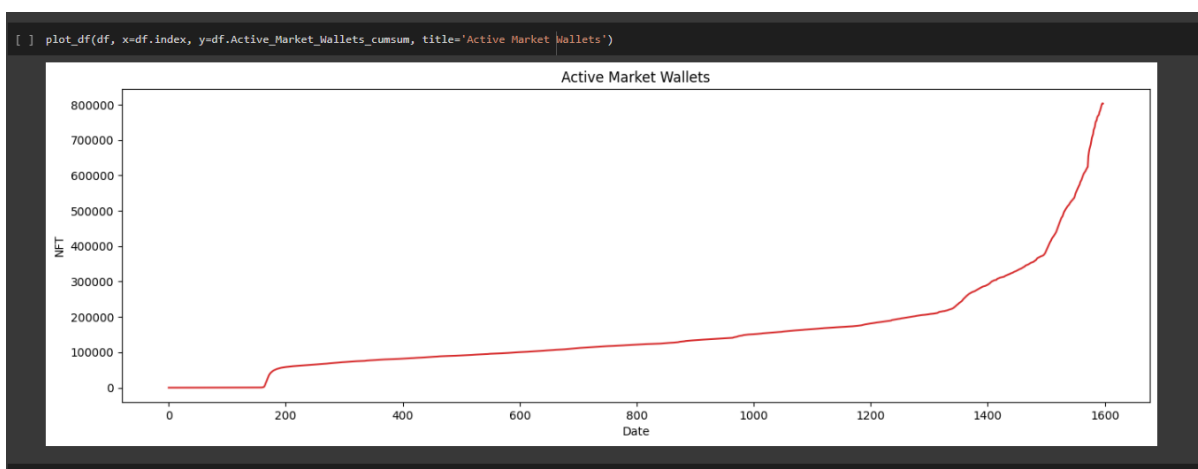
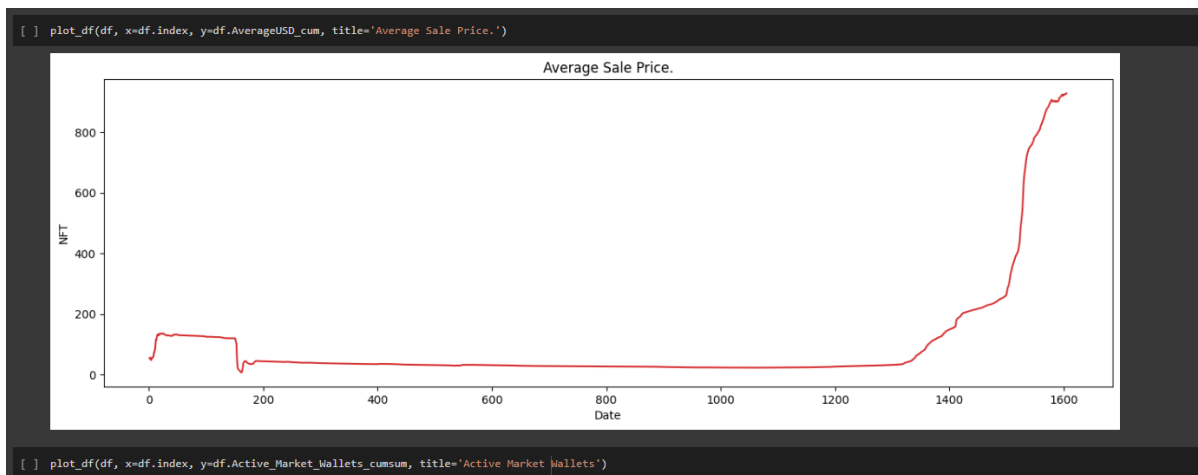
   Date Sales_USD_cumsum Number_of_Sales_cumsum Active_Market_Wallets_cumsum Primary_Sales_cumsum Secondary_Sales_cumsum AverageUSD_cum Sales_USD Number_of_Sales Active_Market_Wallets
0 2017-06-22  0.000000e+00         0          0.000000         0          2749.373832         NaN  6.372813e+06      6866.409346          0
1 2017-06-23  1.020300e+03        19          8.000000         0          19.000000         53.70  1.020300e+03      19.000000          0
2 2017-06-24  2.261140e+03        40         21.000000         0          21.000000         56.53  1.240840e+03      21.000000          0
3 2017-06-25  2.778690e+03        53         28.000000         0          13.000000         52.43  5.175500e+02      13.000000          0
```

```
[ ] import matplotlib.pyplot as plt

# Draw Plot
def plot_df(df, x, y, title="", xlabel='Date', ylabel='NFT', dpi=100):
    plt.figure(figsize=(16,5), dpi=dpi)
    plt.plot(x, y, color='tab:red')
    plt.gca().set(title=title, xlabel=xlabel, ylabel=ylabel)
    plt.show()

plot_df(df, x=df.index, y=df.Number_of_Sales, title='Daily NFT Sales.')
```





## Step 2 : User Interface

```
from tkinter import*
from tkinter import ttk
import tkinter as tk
from PIL import Image,ImageTk
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
```

```
class nft_analytics:
    def __init__(self,root):
        self.root = root
        self.root.geometry("1500x790+0+0")
        self.root.title("Face Recognition System")
```

```

# first image

img = Image.open(r"C:\Users\snehal\Desktop\python project\img\ap.jpg")
img = img.resize((500,130),Image.ANTIALIAS)
self.photoimg = ImageTk.PhotoImage(img)

f_lbl = Label(self.root,image=self.photoimg)
f_lbl.place(x=0,y=0,width=500,height=130)

# second image

img1 = Image.open(r"C:\Users\snehal\Desktop\python
project\img\apsit.jpg")
img1 = img1.resize((500,130),Image.ANTIALIAS)
self.photoimg1 = ImageTk.PhotoImage(img1)

f_lbl = Label(self.root,image=self.photoimg1)
f_lbl.place(x=500,y=0,width=500,height=130)
# third image
img2 = Image.open(r"C:\Users\snehal\Desktop\python project\img\itsa.jpg")
img2 = img2.resize((500,130),Image.ANTIALIAS)
self.photoimg2 = ImageTk.PhotoImage(img2)

f_lbl = Label(self.root,image=self.photoimg2)
f_lbl.place(x=1000,y=0,width=500,height=130)

title_lbl = Label(self.root, text ="NFT SALES ANALYTICS
DASHBOARD" , font =("times new roman",35,"bold"),fg="black")
title_lbl.place(x=0,y=130,width=1530,height =45)
nft_frame=LabelFrame(self.root ,bd =2 , relief = RIDGE , text = "NFT sales
visualization" , font=("times new roman",12,"bold"))
nft_frame.place(x=5,y=200,width=1485,height=300)

btn_frame=Frame(nft_frame , bd=2,relief=RIDGE,bg="white")
btn_frame.place(x=0,y=200,width=1300,height=135)
no_of_sales_btn=Button(btn_frame , text="NUMBER OF SALES"
,command=self.nft ,width=25 ,font =("times new roman",13,"bold"),bg ="white")
no_of_sales_btn.grid(row=0,column=0, padx=25)

```

```

        average_btn=Button(btn_frame,text="AVERAGE SALE PRICE"
,command=self.nft_2,width =25,font=("times new roman",13,"bold"),bg ="white")
        average_btn.grid(row=0,column=1,padx=25)
        market_btn=Button(btn_frame,text="ACTIVE MARKET
WALLETS",command=self.nft_3,width =25,font=("times new roman",13,"bold"),bg
="white")
        market_btn.grid(row=0,column=2,padx=25)
        sale_btn=Button(btn_frame,text="ACTIVE SALES"
,
command=self.nft_4,width =25,font=("times new roman",13,"bold"),bg ="white")
        sale_btn.grid(row=0,column=3,padx=25)
    def nft(self):
        df=pd.read_csv("C:/Users/snehal/Desktop/NFT-Sales-
Analytics/NFT_Sales.csv")
        def plot_df(df, x, y, title="", xlabel='Date', ylabel='NFT', dpi=100):
            figure=plt.figure(figsize=(16,8), dpi=dpi)
            plt.plot(x, y, color='tab:red')
            plt.gca().set(title=title, xlabel=xlabel, ylabel=ylabel)
            line2 = FigureCanvasTkAgg(figure, root)
            line2.get_tk_widget().pack(side=tk.LEFT, fill=tk.BOTH,)
            plot_df(df, x=df.index, y=df.Number_of_Sales, title='Daily NFT Sales.')
    def nft_2(self):
        df=pd.read_csv("C:/Users/snehal/Desktop/NFT-Sales-
Analytics/NFT_Sales.csv")
        def plot_df2(df, x, y, title="", xlabel='Date', ylabel='NFT', dpi=100):
            figure=plt.figure(figsize=(16,8), dpi=dpi)
            plt.plot(x, y, color='tab:red')
            plt.gca().set(title=title, xlabel=xlabel, ylabel=ylabel)
            line2 = FigureCanvasTkAgg(figure, root)
            line2.get_tk_widget().pack(side=tk.LEFT, fill=tk.BOTH,)
            plot_df2(df, x=df.index, y=df.AverageUSD_cum, title='Average Sale
Price.')
    def nft_3(self):
        df=pd.read_csv("C:/Users/snehal/Desktop/NFT-Sales-Analytics/NFT_Sales.csv")
        def plot_df3(df, x, y, title="", xlabel='Date', ylabel='NFT', dpi=100):
            figure=plt.figure(figsize=(16,8), dpi=dpi)
            plt.plot(x, y, color='tab:red')
            plt.gca().set(title=title, xlabel=xlabel, ylabel=ylabel)
            line2 = FigureCanvasTkAgg(figure, root)
            line2.get_tk_widget().pack(side=tk.LEFT, fill=tk.BOTH,)

```



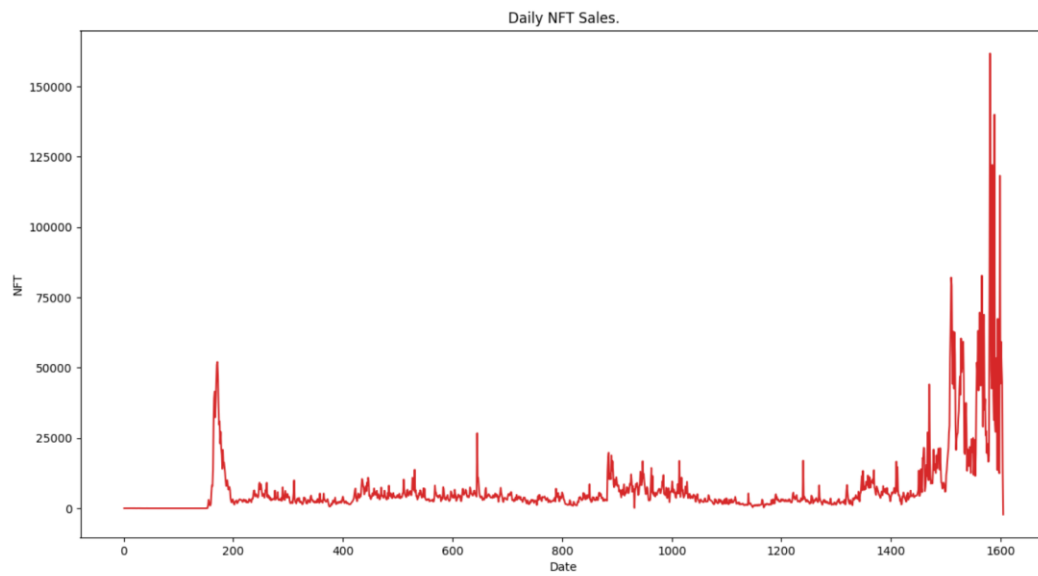
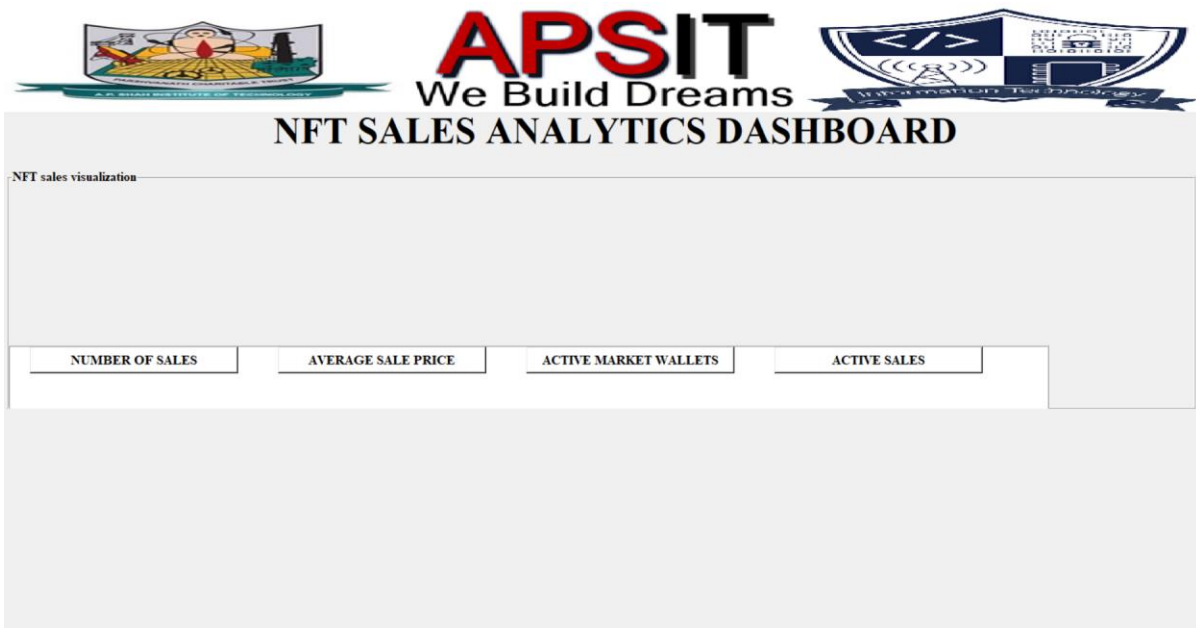
```

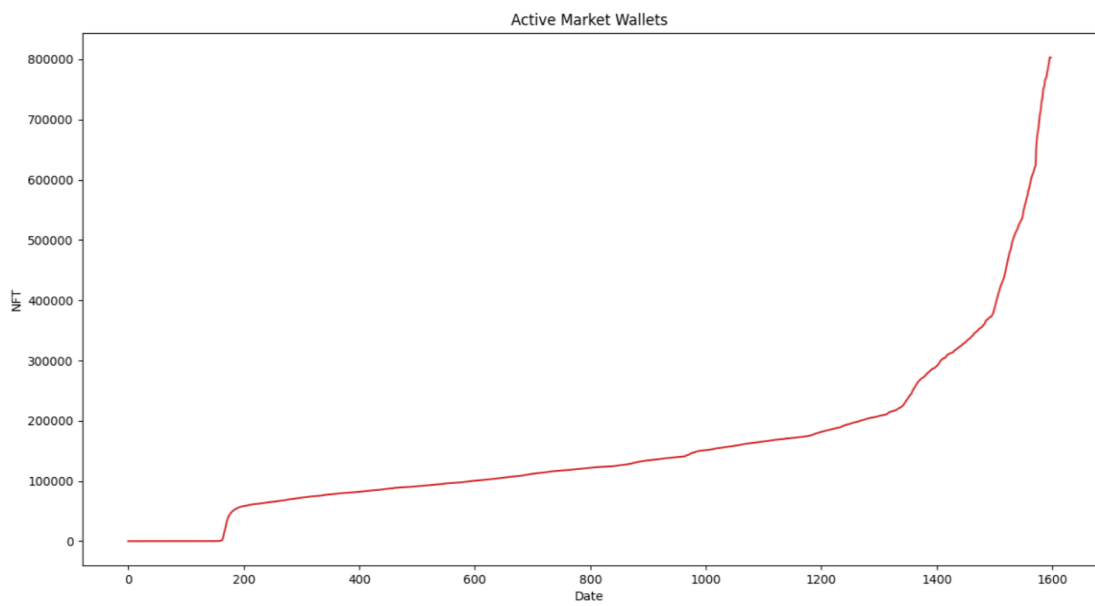
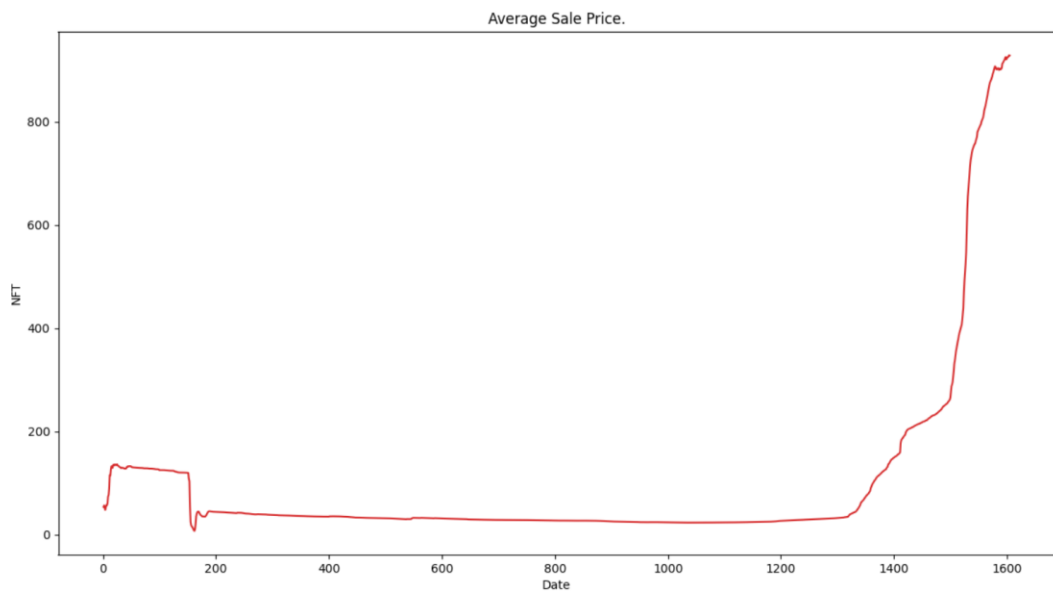
        plot_df3(df,      x=df.index,      y=df.Active_Market_Wallets_cumsum,
title='Active Market Wallets')
    def nft_4(self):

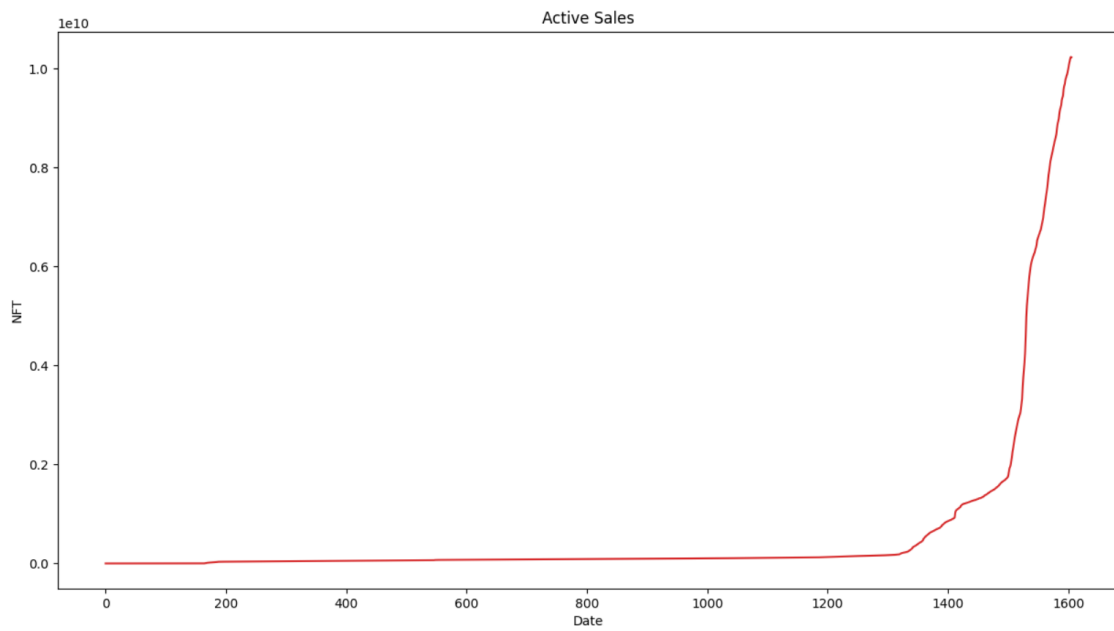
        df=pd.read_csv("C:/Users/snehal/Desktop/NFT-Sales-
Analytics/NFT_Sales.csv")
        def plot_df4(df, x, y, title="", xlabel='Date', ylabel='NFT', dpi=100):
            figure=plt.figure(figsize=(16,8), dpi=dpi)
            plt.plot(x, y, color='tab:red')
            plt.gca().set(title=title, xlabel=xlabel, ylabel=ylabel)
            line2 = FigureCanvasTkAgg(figure, root)
            line2.get_tk_widget().pack(side=tk.LEFT, fill=tk.BOTH,)
        plot_df4(df, x=df.index, y=df.Sales_USD_cumsum, title='Active Sales')
if __name__ == "__main__":
    root = tk.Tk()
    obj = nft_analytics(root)
    root.mainloop()

```

## 7. RESULTS







IBM Cognos Analytics with Watson

\* New data module

Search content

Properties

Data module

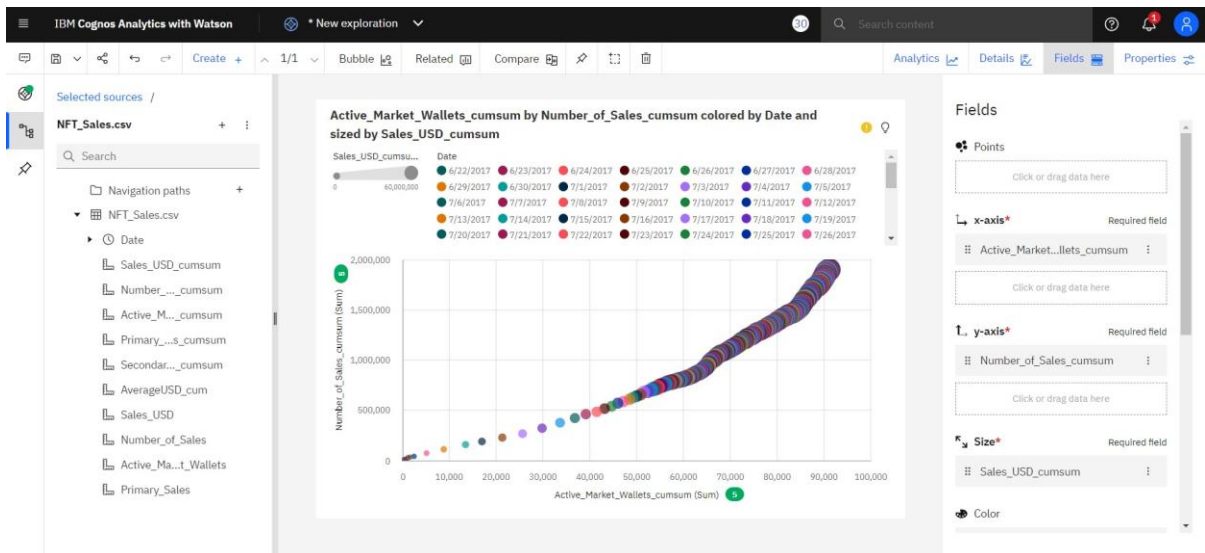
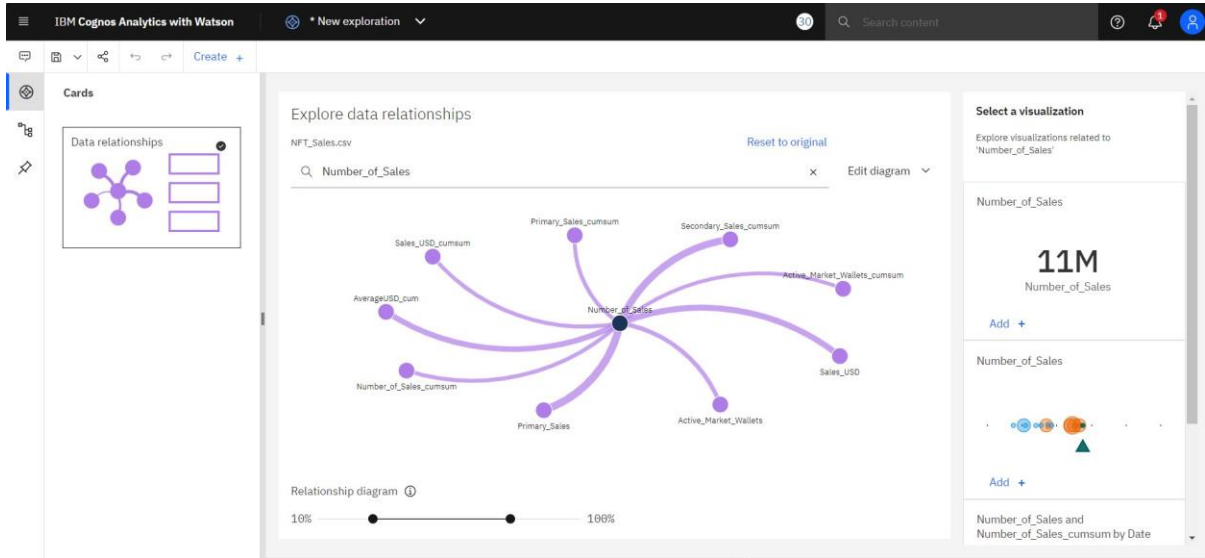
Grid Relationships Custom tables

Row Id	Date	Sales_USD_cumsum	Number_of...es_cumsum	Active_Mar...ts_cumsum	Primary_Sales_cumsum	Secondary...es_c
1	2017-06-22	0	0	0	0	Null
2	2017-06-23	1020.3	19	8	0	19
3	2017-06-24	2261.14	40	21	0	21
4	2017-06-25	2778.69	53	28	0	13
5	2017-06-26	3203.32	67	34	0	14
6	2017-06-27	5296.23	100	44	0	33
7	2017-06-28	6543.49	115	46	0	15
8	2017-06-29	8729.21	153	49	0	38
9	2017-06-30	10437.36	174	53	0	21
10	2017-07-01	13578.16	184	55	0	10
11	2017-07-02	16502.77	215	57	0	31
12	2017-07-03	23752.56	266	59	0	51
13	2017-07-04	37258.49	324	62	0	58

Navigation paths

- NFT\_Sales.csv
  - # Row Id
  - 🕒 Date
  - Sales\_USD\_cumsum
  - Number\_of...s\_cumsum
  - Active\_M...s\_cumsum
  - Primary...s\_cumsum
  - Secondar...s\_cumsum
  - AverageUSD\_cum
  - Sales\_USD
  - Number\_of\_Sales
  - Active\_Mar...t\_Wallets
  - Primary\_Sales





## **8. CONCLUSION**

The Global Non-fungible Token Market size is expected to reach \$97.6 billion by 2028, rising at a market growth of 31.6% CAGR during the forecast period. Non-fungible tokens (NFTs) are blockchain-based cryptographic assets having unique metadata and identification codes that separate them from one another. Here we performed visualization tasks on the NFT sales analytics dataset available on Kaggle and analyzed the data to gain insights through the front end created using Tkinter. Preprocessing tasks were performed in order to remove noise and make data clean.

## **9. FUTURE SCOPE**

The future scope of the NFT Analytics Dashboard is to provide a platform for users to get insights about their NFTs. Users will be able to view information about their NFTs including;

- Total Supply
- Price History
- Market Cap
- Volume History
- Top Buyers
- Top Sellers



## REFERENCES

- A. Park, J. Kietzmann, L. Pitt and A. Dabirian, "The Evolution of Non Fungible Tokens: Complexity and Novelty of NFT Use-Cases," in IT Professional, vol. 24, no. 1, pp. 9-14, 1 Jan.-Feb. 2022, doi: 10.1109/MITP.2021.3136055.
- Mapping the NFT revolution: market trends, trade networks, and visual features
- S. Bouraga, "On the Popularity of Non-Fungible Tokens: Preliminary Results," 2021 3rd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), 2021, pp. 49-50, doi: 10.1109/BRAINS52497.2021.9569792.
- S. Bradić, D. Delija, G. Sirovatka and M. Žagar, "Creating own NFT token using erc721 standard and solidity programming language," 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), 2022, pp. 1053-1056, doi: 10.23919/MIPRO55190.2022.9803593.