**A PROJECT REPORT**

**ON**

**WHATSAPP GROUP CHAT ANALYSIS**

**(USING PYTHON)**



Submitted in the partial fulfillment of the requirements for the award of the Degree of **Bachelor of Computer Science**

**SUBMITTED BY**

**PAGI AKASH SADANAND**

**Under The Guidance of Prof. Swetha Iyer**

**UNIVERSITY OF MUMBAI**

**T.Y.BSC. (COMPUTER SCIENCE)**

**(ACADEMIC YEAR 2022-23)**

**DECLARATION**

I hereby declare that I hereby declare that **Pagi Akash Sadanand** of T.Y.BSC Computer Science, **Roll No .19** this project work entitled “**WHATSAPP GROUP CHAT ANALYSIS(Using Python)**” has been prepared by me during the year 2022 – 23 under the guidance of Prof. Swetha Iyer, Department of Computer Science, Vartak College, Vasai in the partial fulfillment of the degree prescribed by the college.

I also declare that this project is the outcome of my own effort, I have not copied from any other students’ work or from any

other sources except where due reference or acknowledgment is made explicitly, nor has any part been authored by another person.

All the documents and the reports submitted are true to the best of my knowledge.

Signature

**ACKNOWLEDGEMENT**

I would like to express my greatest appreciation to the all individuals who have helped and supported me throughout the project. I am thankful to my computer teacher for his ongoing support during the project, from initial advice, and encouragement, which led to the final report of this project. I would also like to thank Prof. Swetha Iyer who was always there in our computer lab for guiding the project.

A special acknowledgement goes to my classmates who helped me in completing the project by exchanging interesting ideas and sharing their experience.

I wish to thank my parents as well for their undivided support and interest who inspired me and encouraged me to go my own way, without whom I would be unable to complete my project.

**INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| **SR.NO** | **CONTENTS** | **PAGE.NO** | **SIGN** |
| 1 | Gantt chart |  |  |
| 2 | Introduction (synopsis) |  |  |
| 3 | Limitations of current system |  |  |
| 4 | Advantage of proposed system |  |  |
| 5 | **Feasibility study:**  1.Technical feasibility  2.Economical feasibility  3.Operational feasibility |  |  |
| 6 | Requirement specifications (software and hardware requirements) |  |  |
| 7 | **System design and details**  1.Event table  2.Entity relationship diagram  3.Class diagram  4.Activity diagram  5.Use case diagram  6.Sequence diagram  7.Component diagram  8.Deployment diagram |  |  |
| 8 | System Implementation (Code) |  |  |
| 9 | User Interface (Screenshot) |  |  |
| 10 | Conclusion and future scope |  |  |
| 11 | References |  |  |

**INTRODUCTION (synopsis)**

WhatsApp Analyzer means we are analyzing our WhatsApp group activities.

It tracks our conversation and analyzes how much time we are spending on WhatsApp

Here I used different python libraries which help me to extract useful information from raw data.

This project uses simple python modules/libraries such as pandas, matplotlib, seaborn , Regex(re) , urlextract, wordcloud which are used to create data frames and plot different graphs, where then it is displayed in the streamlit application.

**LIMITATIONS OF CURRENT SYSTEM**

**1.Software limitations:**

The WhatsApp group chat analyzer software may have limitations in terms of the amount of data it can analyze or the types of data it can process. Some software may not be able to handle large group chats or may not be able to analyze media files such as photos and videos.

**2.Connectivity limitations:**

If you are analyzing a WhatsApp group chat that has a large amount of data, it may take a long time to upload and download the data from the WhatsApp servers. Slow internet connectivity may also affect the performance of the software.

**3.Privacy limitations:**

Depending on the settings of the WhatsApp group chat analyzer software, it may not be able to access certain data or may not be able to provide detailed information about individual users in the chat. This could limit the usefulness of the analysis.

**ADVANTAGES OF PROPOSED SYSTEM**

### **Improved decision making:**

### whatsapp group chat analysis can help organizations make better decisions by providing insights into customer behavior, employee productivity, and other key factors that impact business performance.

### **Enhanced communication:**

### whatsapp group chat analysis can help organizations improve communication and collaboration by identifying patterns in communication and highlighting areas where communication could be improved.

### **Cost savings:**

### whatsapp group chat analysis can help organizations identify areas where they can save costs by improving efficiency, reducing waste, and streamlining processes.

### **FEASIBILITY STUDY**

**1.TECHNICAL FEASIBILITY :**

Technical feasibility for WhatsApp group chat analysis involves assessing the technical requirements, capabilities, and limitations of conducting the analysis. Here are some key factors to consider when evaluating the technical feasibility of WhatsApp group chat analysis:

Data Collection: It is important to consider the availability, accessibility, and quality of data, as well as the methods for collecting data.

Data Analysis: This could involve evaluating the data analysis tools and techniques, as well as the computational resources required to conduct the analysis.

Technical feasibility for WhatsApp group chat analysis involves evaluating the feasibility of collecting, storing, and analyzing data, as well as the hardware, software, and skills required to conduct the analysis. It is important to consider the technical feasibility of the project to ensure that it can be carried out effectively and efficiently.

**2. ECONOMICAL FEASIBILITY:**

Economic feasibility for WhatsApp group chat analysis involves evaluating the economic feasibility of WhatsApp group chat analysis:

Costs: it includes the cost of data collection and analysis tools, hardware and software, training and support, and any other resources required to conduct the analysis.

Benefits: it could includes the potential improvements in organizational efficiency, productivity, decision-making, and customer satisfaction resulting from the analysis.

Overall, economic feasibility for WhatsApp group chat analysis involves evaluating the costs and benefits associated with the project and determining whether the benefits outweigh the costs. It is important to consider the economic feasibility of the project to ensure that it is financially viable and sustainable over the long term.

**3. OPERATIONAL FEASIBILITY:**

Operational feasibility for WhatsApp group chat analysis involves assessing whether the proposed system meets the user requirements, is compatible with the organization's existing systems and processes, is feasible to implement, and is sustainable over the long term. It is important to consider the operational feasibility of the project to ensure that the proposed system can be effectively integrated into the organization's operations and meets the needs of the users.

**REQUIREMENT SPECIFICATION**

* **HARDWARE REQUIREMENT :**

**Processor:** Intel i3 or above or Ryzen 3 or above

**Hard Disks:** 500GB HDD or 256 SSD or More

**RAM:** 2GB or More

* **SOFTWARE REQUIREMENT :**

**OS :** Windows 8/10/11 or Linux or MacOS

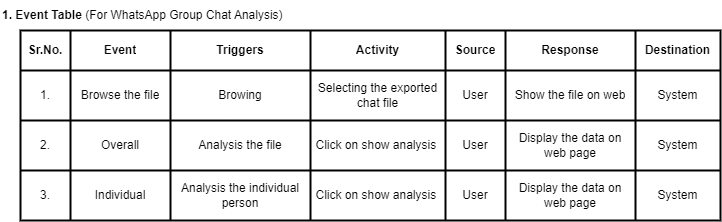
**IDE :** VSCode & Jupyter notebook (for testing and execution of the code)

**Language :** Python v3.10.0

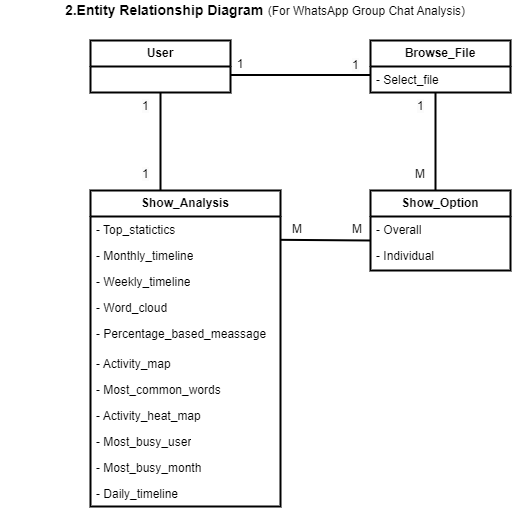
**Libraries/Modules :** streamlit (Framework) ,Regex ,matplotlib ,seaborn ,urlextract ,wordcloud , pandas **.**

**SYSTEM DESIGN**

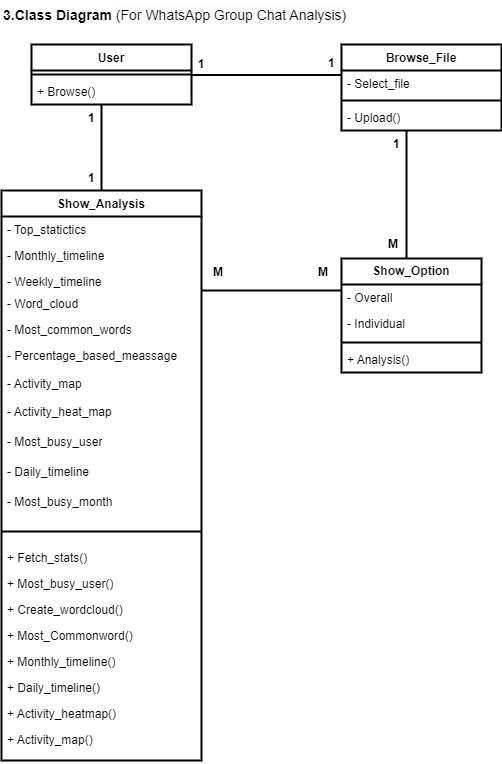
**1.EVENT TABLE:**

****

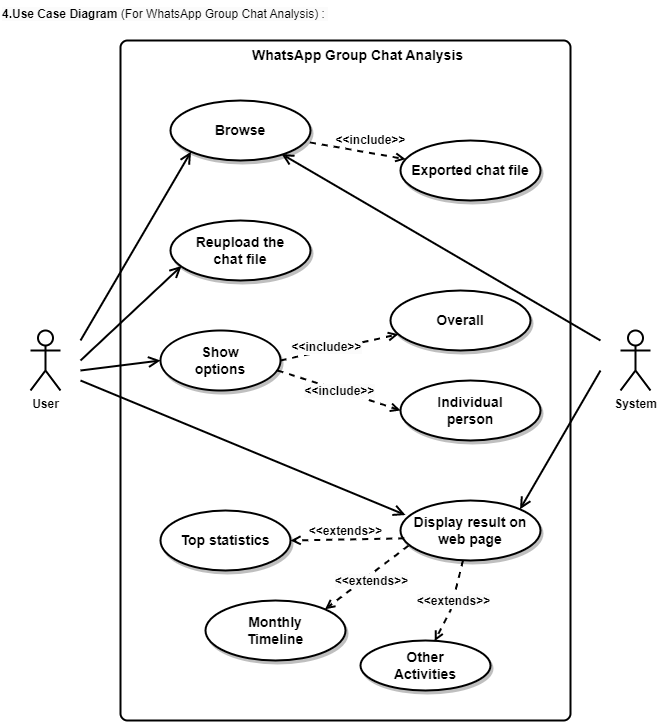
**2.ENTITY RELATIONSHIP DIAGRAM :**

****

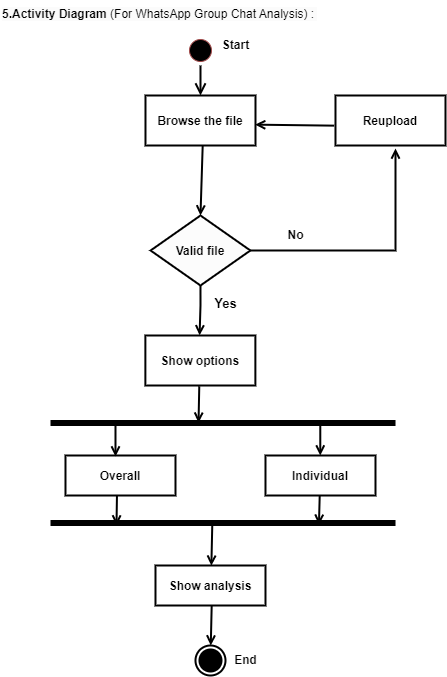
**3.CLASS DIAGRAM :**

****

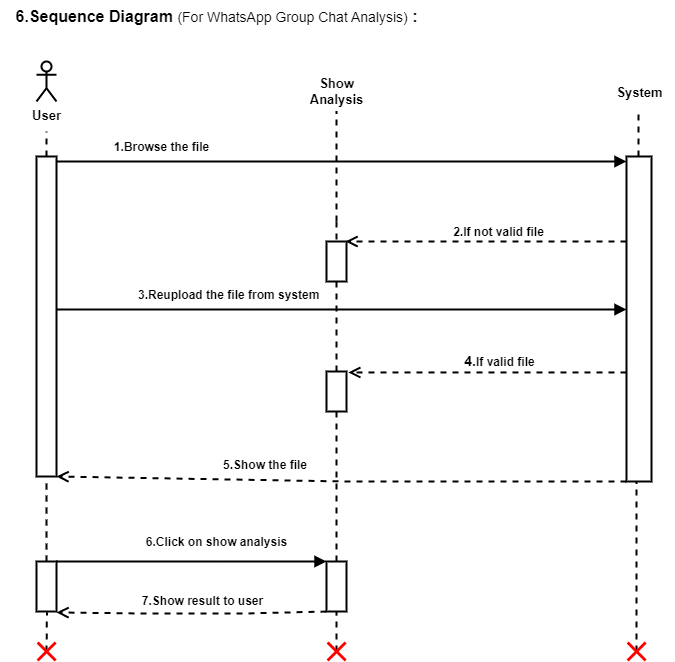
**4.USE CASE DIAGRAM:**

****

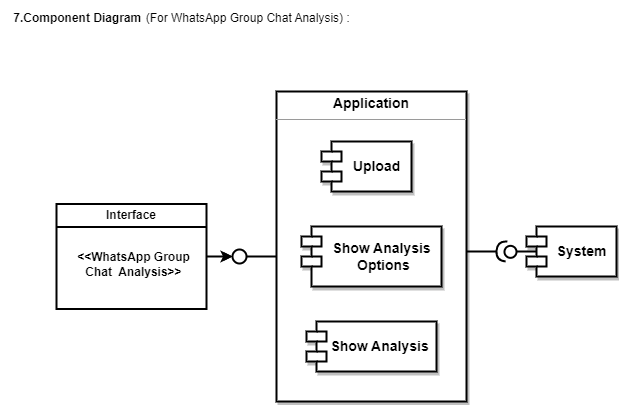
**5.ACTIVITY DIAGRAM:**

****

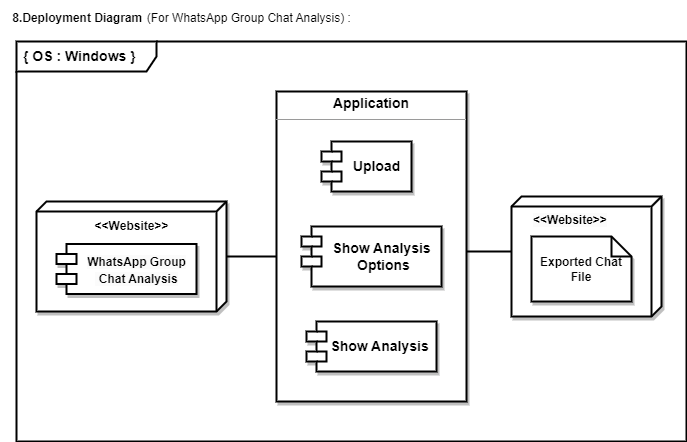
**6.SEQUENCE DIAGRAM:**

****

**7.COMPONENT DIAGRAM:**

****

**8.DEPLOYMENT DIAGRAM:**

****

**SYSTEM IMPLEMENTATION (CODE)**

**# app.py :**

import streamlit as st

import preprocessor,helper

import matplotlib.pyplot as plt

import seaborn as sns

st.sidebar.title("Whatsapp Group Chat Analysis")

uploaded\_file = st.sidebar.file\_uploader("Choose a file")

if uploaded\_file is not None:

bytes\_data = uploaded\_file.getvalue()

data = bytes\_data.decode("utf-8")

df = preprocessor.preprocess(data)

# fetch unique users

user\_list = df['user'].unique().tolist()

user\_list.remove('group\_notification')

user\_list.sort()

user\_list.insert(0,"Overall")

selected\_user = st.sidebar.selectbox("Show analysis wrt",user\_list)

if st.sidebar.button("Show Analysis"):

# Stats Area

num\_messages, words, num\_media\_messages, num\_links = helper.fetch\_stats(selected\_user,df)

st.title("Top Statistics")

col1, col2, col3, col4 = st.columns(4)

with col1:

st.subheader("Total Messages")

st.title(num\_messages)

with col2:

st.subheader("Total Words")

st.title(words)

with col3:

st.subheader("Media Shared")

st.title(num\_media\_messages)

with col4:

st.subheader("Links Shared")

st.title(num\_links)

# monthly timeline

st.title("Monthly Timeline")

timeline = helper.monthly\_timeline(selected\_user,df)

fig,ax = plt.subplots()

ax.plot(timeline['time'], timeline['message'],color='green')

plt.xticks(rotation='vertical')

st.pyplot(fig)

# daily timeline

st.title("Daily Timeline")

daily\_timeline = helper.daily\_timeline(selected\_user, df)

fig, ax = plt.subplots()

ax.plot(daily\_timeline['only\_date'], daily\_timeline['message'], color='orange')

plt.xticks(rotation='vertical')

st.pyplot(fig)

# activity map

st.title('Activity Map')

col1,col2 = st.columns(2)

with col1:

st.subheader("Most busy day")

busy\_day = helper.week\_activity\_map(selected\_user,df)

fig,ax = plt.subplots()

ax.bar(busy\_day.index,busy\_day.values,color='purple')

plt.xticks(rotation='vertical')

st.pyplot(fig)

with col2:

st.subheader("Most busy month")

busy\_month = helper.month\_activity\_map(selected\_user, df)

fig, ax = plt.subplots()

ax.bar(busy\_month.index, busy\_month.values,color='lightgreen')

plt.xticks(rotation='vertical')

st.pyplot(fig)

st.title("> Weekly Activity Map (Heatmap)")

user\_heatmap = helper.activity\_heatmap(selected\_user,df)

fig,ax = plt.subplots()

ax = sns.heatmap(user\_heatmap)

st.pyplot(fig)

# finding the busiest users in the group(Group level)

if selected\_user == 'Overall':

st.title('Most Busy Users')

x,new\_df = helper.most\_busy\_users(df)

fig, ax = plt.subplots()

plt.xlabel("Group users")

plt.ylabel("Messages")

col1, col2 = st.columns(2)

with col1:

ax.bar(x.index, x.values,color='skyblue')

plt.xticks(rotation='vertical')

st.pyplot(fig)

with col2:

st.subheader('Percentages Based Messages ')

st.dataframe(new\_df)

# WordCloud

st.title("Wordcloud")

df\_wc = helper.create\_wordcloud(selected\_user,df)

fig,ax = plt.subplots()

ax.imshow(df\_wc)

st.pyplot(fig)

# most common words

st.title('Most common words')

most\_common\_df = helper.most\_common\_words(selected\_user,df)

fig,ax = plt.subplots()

plt.xlabel("Meassages (words) count")

plt.ylabel("Common words in chat")

ax.barh(most\_common\_df[0],most\_common\_df[1])

plt.xticks(rotation='vertical')

st.pyplot(fig)

**# helper.py**

import streamlit as st

import preprocessor,helper

import matplotlib.pyplot as plt

import seaborn as sns

st.sidebar.title("Whatsapp Group Chat Analysis")

uploaded\_file = st.sidebar.file\_uploader("Choose a file")

if uploaded\_file is not None:

bytes\_data = uploaded\_file.getvalue()

data = bytes\_data.decode("utf-8")

df = preprocessor.preprocess(data)

# fetch unique users

user\_list = df['user'].unique().tolist()

user\_list.remove('group\_notification')

user\_list.sort()

user\_list.insert(0,"Overall")

selected\_user = st.sidebar.selectbox("Show analysis wrt",user\_list)

if st.sidebar.button("Show Analysis"):

# Stats Area

num\_messages, words, num\_media\_messages, num\_links = helper.fetch\_stats(selected\_user,df)

st.title("Top Statistics")

col1, col2, col3, col4 = st.columns(4)

with col1:

st.subheader("Total Messages")

st.title(num\_messages)

with col2:

st.subheader("Total Words")

st.title(words)

with col3:

st.subheader("Media Shared")

st.title(num\_media\_messages)

with col4:

st.subheader("Links Shared")

st.title(num\_links)

# monthly timeline

st.title("Monthly Timeline")

timeline = helper.monthly\_timeline(selected\_user,df)

fig,ax = plt.subplots()

ax.plot(timeline['time'], timeline['message'],color='green')

plt.xticks(rotation='vertical')

st.pyplot(fig)

# daily timeline

st.title("Daily Timeline")

daily\_timeline = helper.daily\_timeline(selected\_user, df)

fig, ax = plt.subplots()

ax.plot(daily\_timeline['only\_date'], daily\_timeline['message'], color='orange')

plt.xticks(rotation='vertical')

st.pyplot(fig)

# activity map

st.title('Activity Map')

col1,col2 = st.columns(2)

with col1:

st.subheader("Most busy day")

busy\_day = helper.week\_activity\_map(selected\_user,df)

fig,ax = plt.subplots()

ax.bar(busy\_day.index,busy\_day.values,color='purple')

plt.xticks(rotation='vertical')

st.pyplot(fig)

with col2:

st.subheader("Most busy month")

busy\_month = helper.month\_activity\_map(selected\_user, df)

fig, ax = plt.subplots()

ax.bar(busy\_month.index, busy\_month.values,color='lightgreen')

plt.xticks(rotation='vertical')

st.pyplot(fig)

st.title("Weekly Activity Map (Heatmap)")

user\_heatmap = helper.activity\_heatmap(selected\_user,df)

fig,ax = plt.subplots()

ax = sns.heatmap(user\_heatmap, linewidth=1, linecolor='black', square=True, cmap='Blues') #annot=True

st.pyplot(fig)

# finding the busiest users in the group(Group level)

if selected\_user == 'Overall':

st.title('Most Busy Users')

x,new\_df = helper.most\_busy\_users(df)

fig, ax = plt.subplots()

plt.xlabel("Group users")

plt.ylabel("Messages")

col1, col2 = st.columns(2)

with col1:

ax.bar(x.index, x.values,color='skyblue')

plt.xticks(rotation='vertical')

st.pyplot(fig)

with col2:

st.subheader('Percentages Based Messages ')

st.dataframe(new\_df)

# WordCloud

st.title("Wordcloud")

df\_wc = helper.create\_wordcloud(selected\_user,df)

fig,ax = plt.subplots()

ax.imshow(df\_wc)

st.pyplot(fig)

# most common words

st.title('Most common words')

most\_common\_df = helper.most\_common\_words(selected\_user,df)

fig,ax = plt.subplots()

plt.xlabel("Meassages (words) count")

plt.ylabel("Common words in chat")

ax.barh(most\_common\_df[0],most\_common\_df[1])

plt.xticks(rotation='vertical')

st.pyplot(fig)

**# preprocessor.py**

import re

import pandas as pd

def preprocess(data):

pattern = '\d{1,2}\/\d{2,4}\/\d{2,4},\s\d{1,2}:\d{1,2}\s\w{1,2}\s-\s'

messages = re.split(pattern, data)[1:]

dates = re.findall(pattern, data)

df = pd.DataFrame({'user\_message': messages, 'message\_date': dates})

# convert message\_date type

df['message\_date'] = pd.to\_datetime(df['message\_date'], format ='%d/%m/%Y, %I:%M %p - ')

df.rename(columns={'message\_date': 'date'}, inplace=True)

users = []

messages = []

for message in df['user\_message']:

entry = re.split('([\w\W]+?):\s', message)

if entry[1:]: # user name

users.append(entry[1])

messages.append(" ".join(entry[2:]))

else:

users.append('group\_notification')

messages.append(entry[0])

df['user'] = users

df['message'] = messages

df.drop(columns=['user\_message'], inplace=True)

df['only\_date'] = df['date'].dt.date

df['year'] = df['date'].dt.year

df['month\_num'] = df['date'].dt.month

df['month'] = df['date'].dt.month\_name()

df['day'] = df['date'].dt.day

df['day\_name'] = df['date'].dt.day\_name()

df['hour'] = df['date'].dt.hour

df['minute'] = df['date'].dt.minute

period = []

for hour in df[['day\_name', 'hour']]['hour']:

if hour == 23:

period.append(str(hour) + "-" + str('00'))

elif hour == 0:

period.append(str('00') + "-" + str(hour + 1))

else:

period.append(str(hour) + "-" + str(hour + 1))

df['period'] = period

return df

**# requirements.txt**

streamlit

matplotlib

seaborn

urlextract

wordcloud

pandas

emoji

**# setup.uh**

mkdir -p ~/.streamlit/

echo "\

[server]\n\

port = $PORT\n\

enableCORS = false\n\

headless = true\n\

\n\

" > ~/.streamlit/config.toml

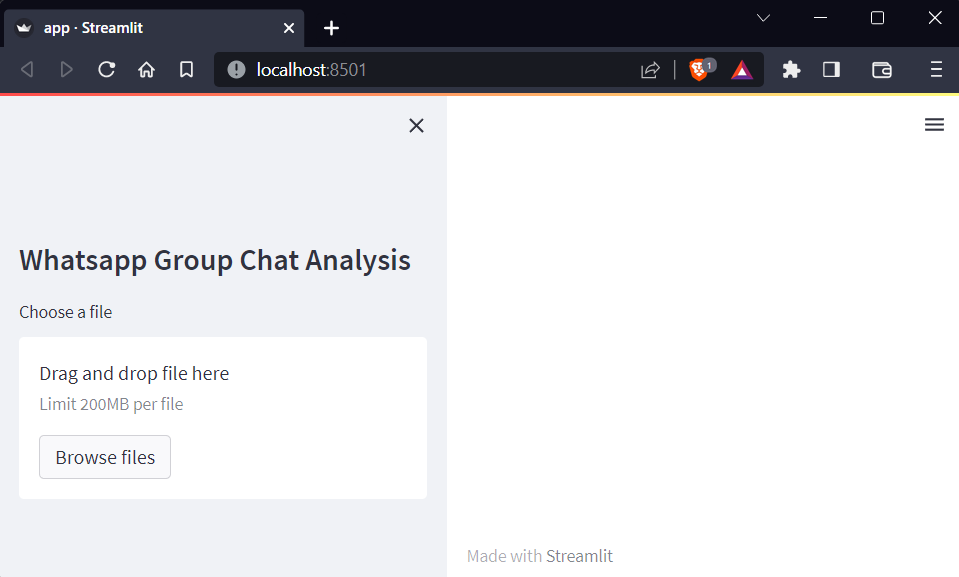
**# stop\_hinglish.txt (file)**

**link : https://github.com/asp977/Whatsapp-Chat-analyzer/blob/main/stop\_hinglish.txt**

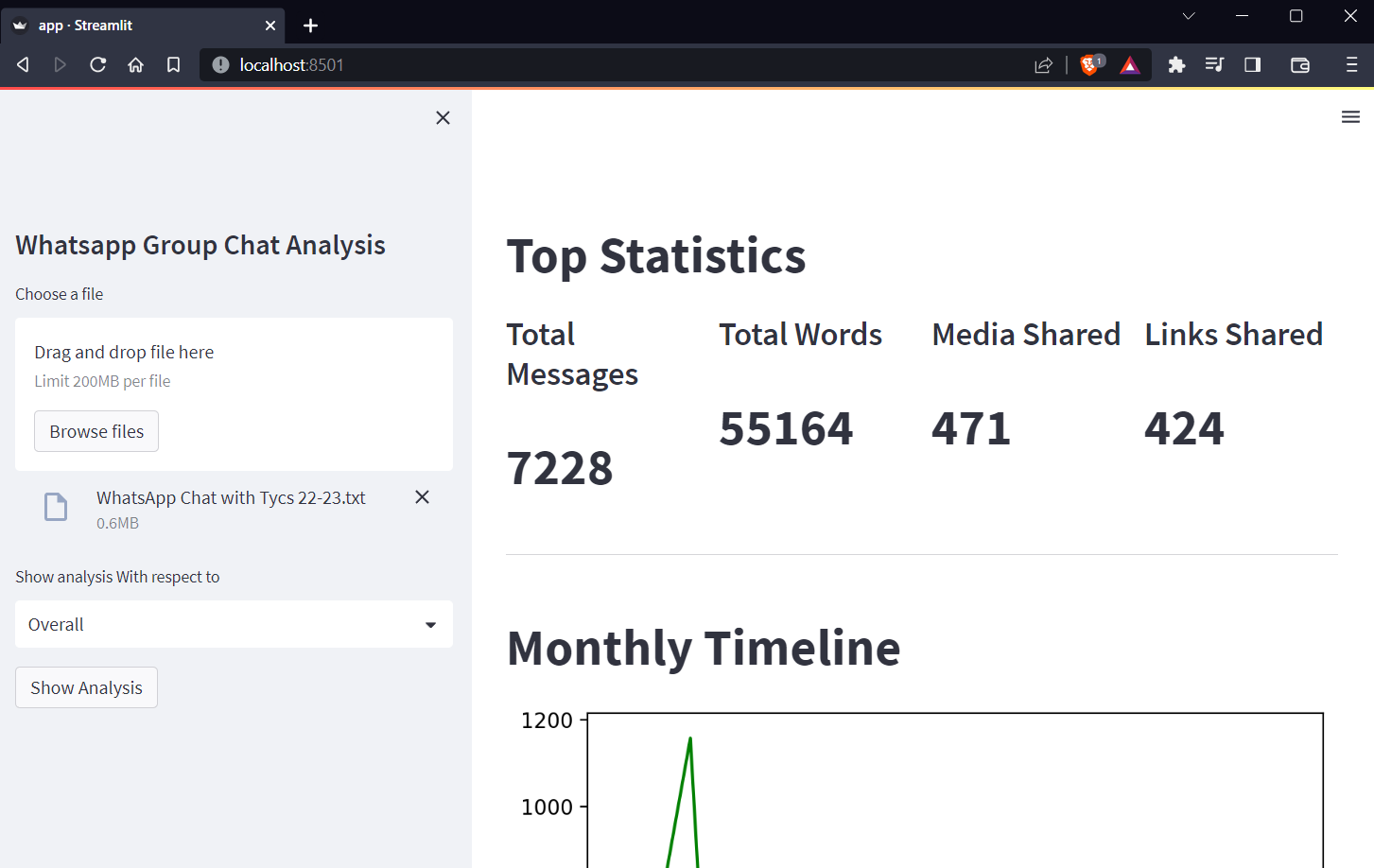
**USER INTERFACE (SCREENSHOTS)**

**# Hosting link:**

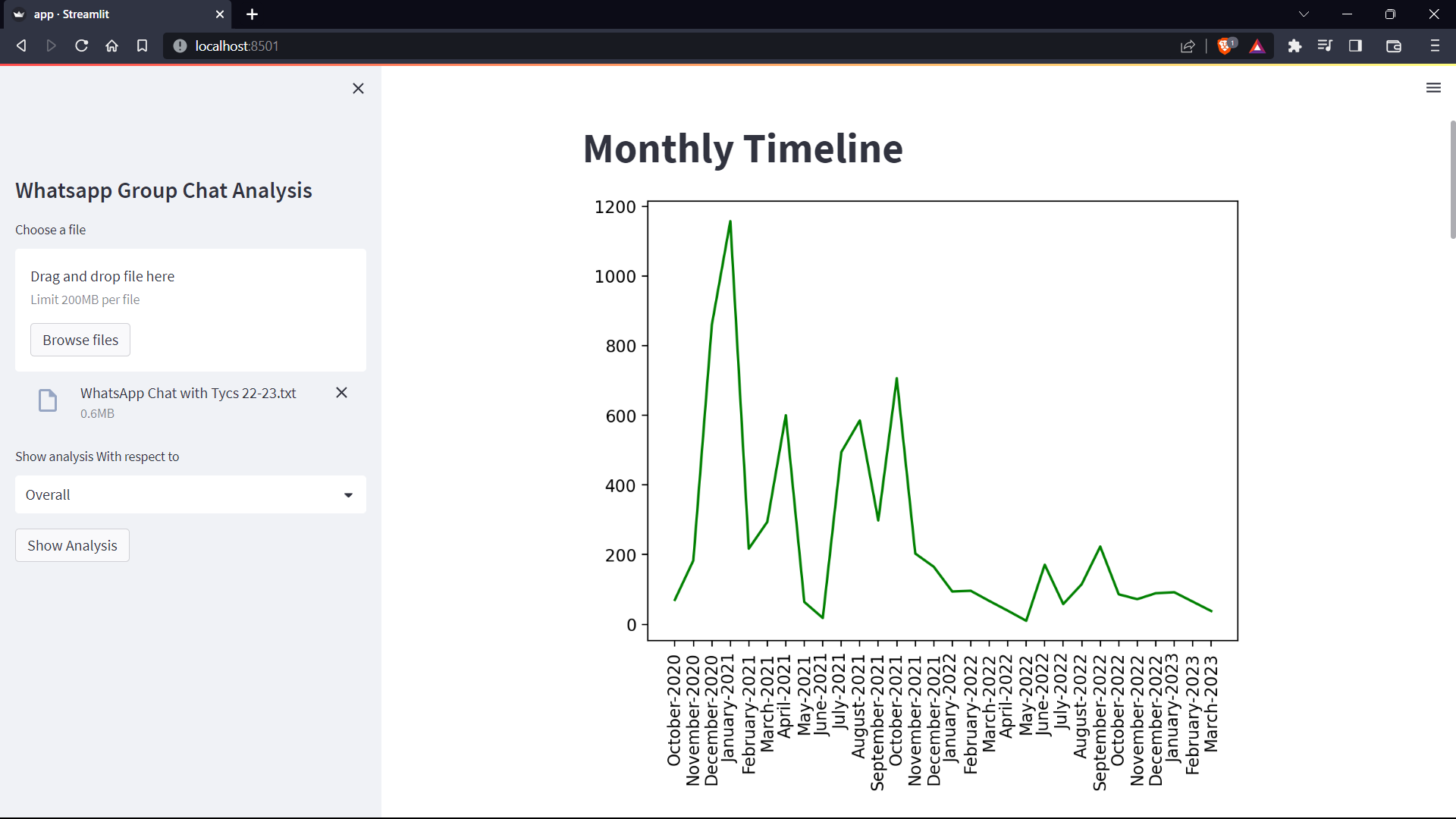
**Default chat file uploading page :**



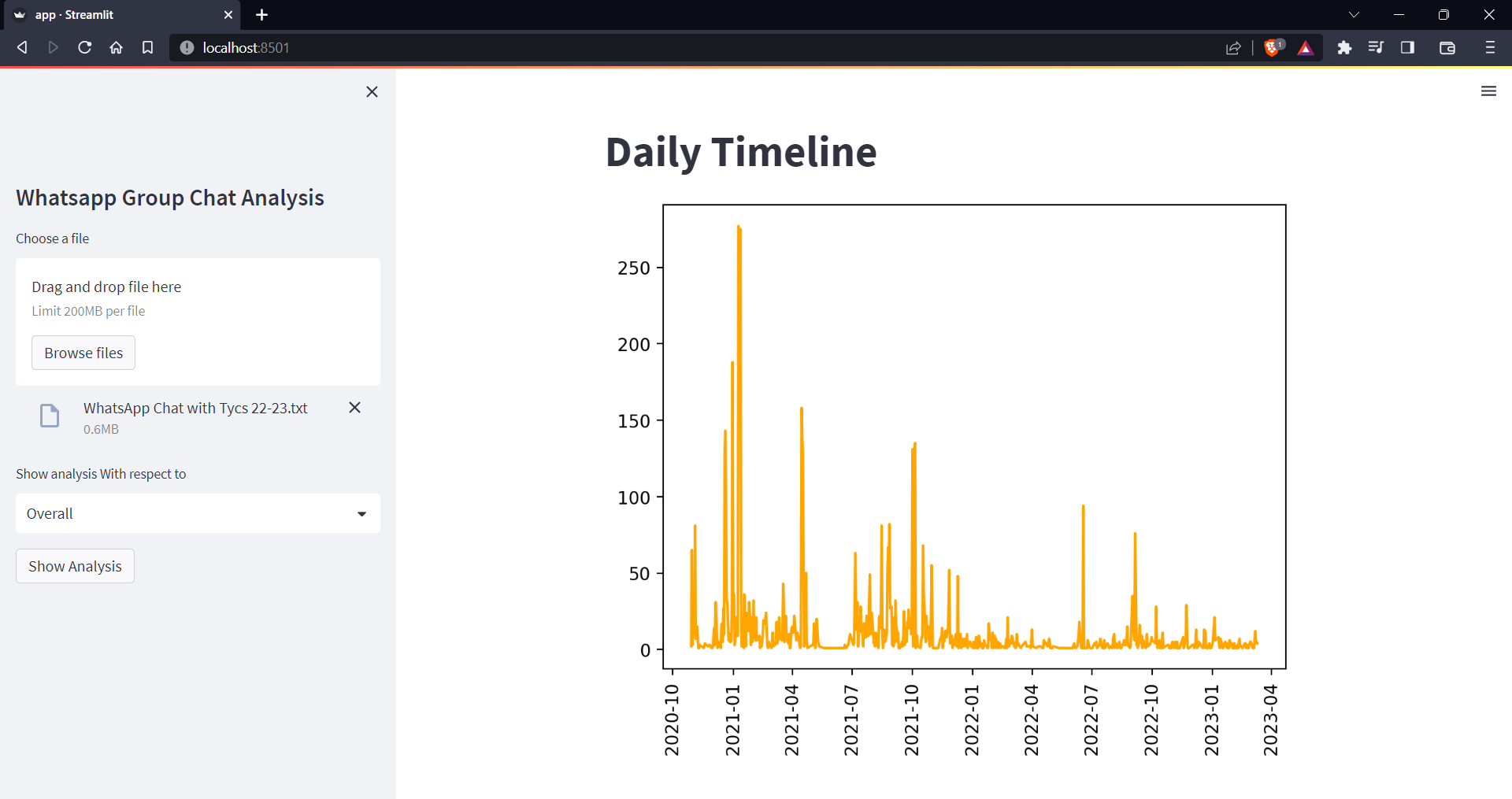
**Top Statistics :**



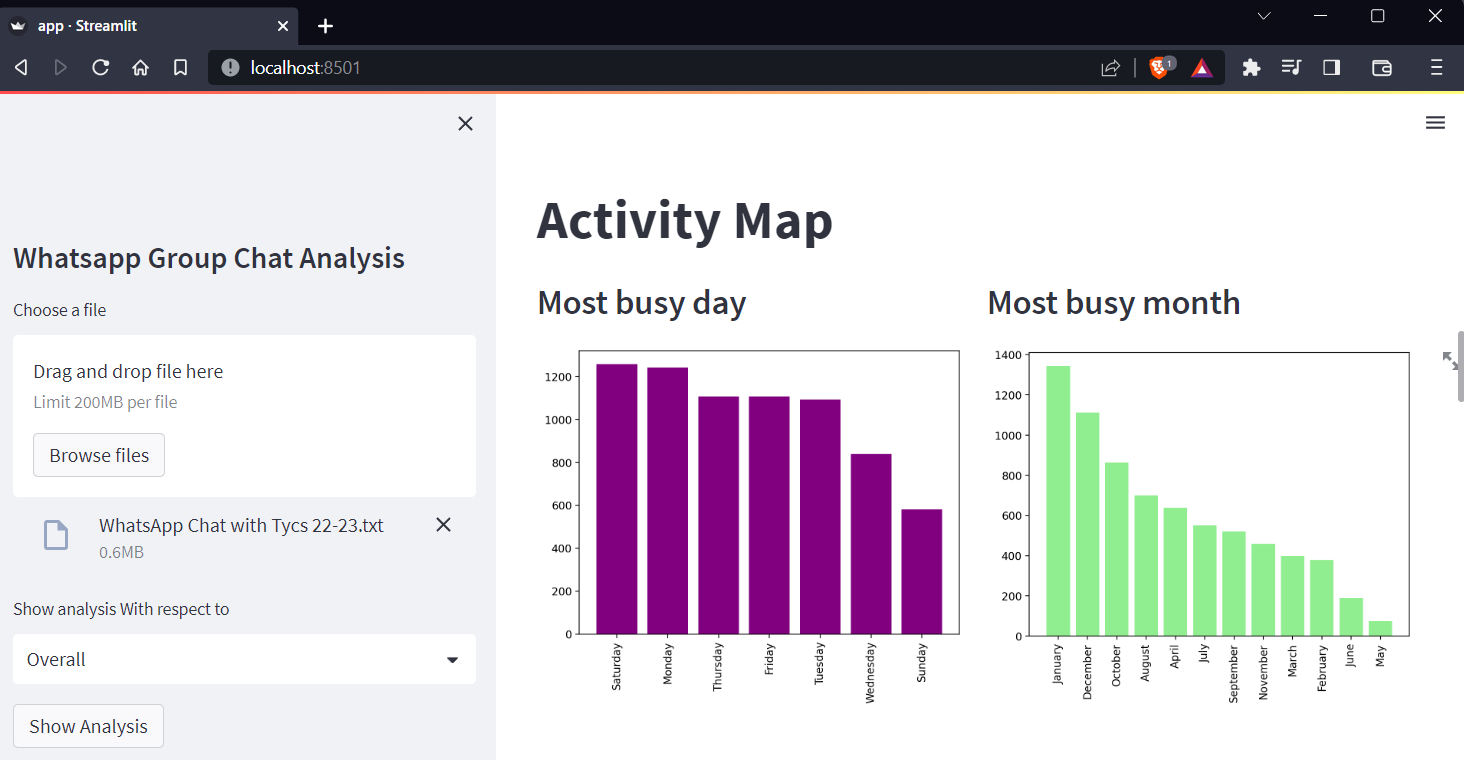
**Monthly Timeline :**



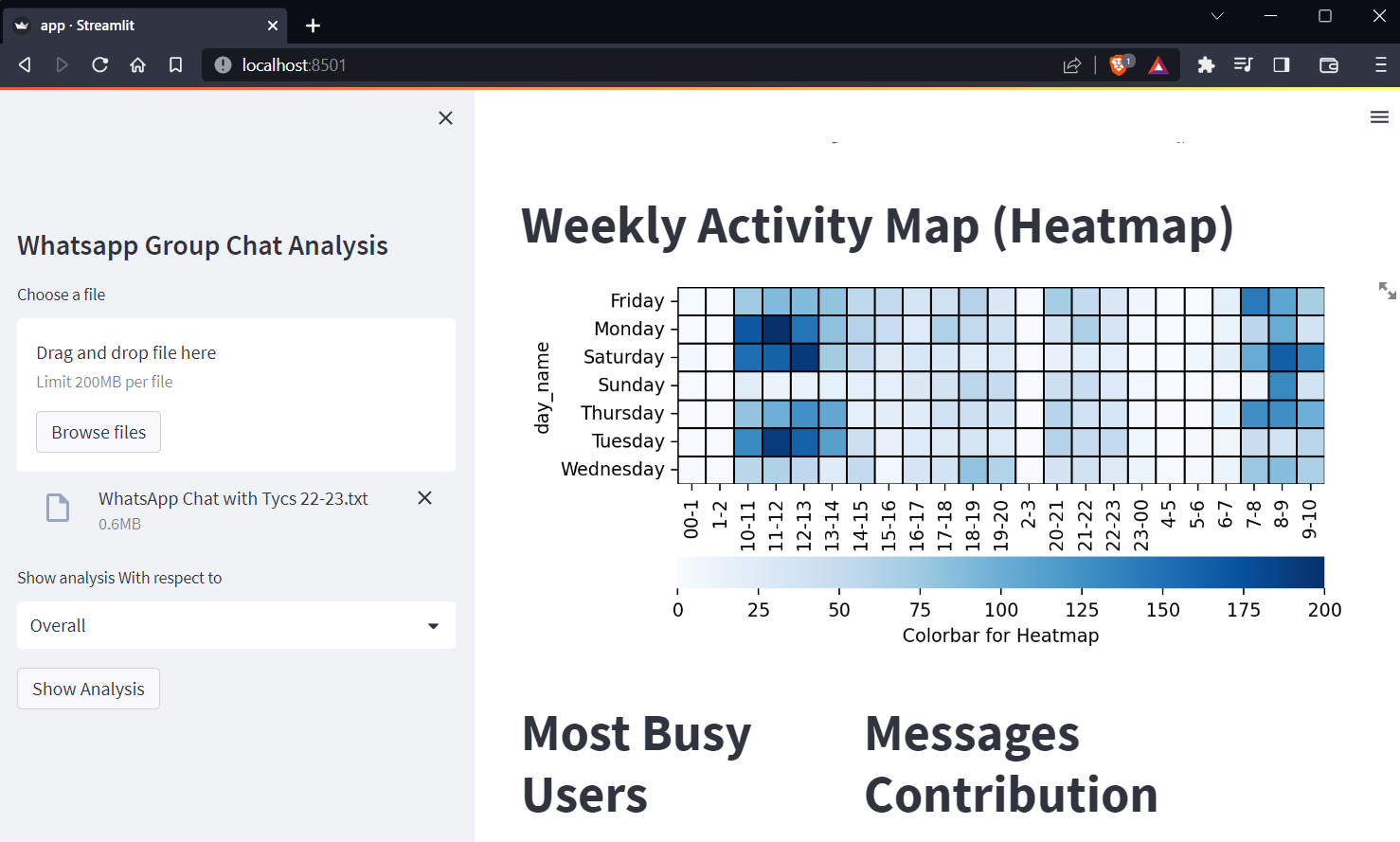
**Daily Timeline :**



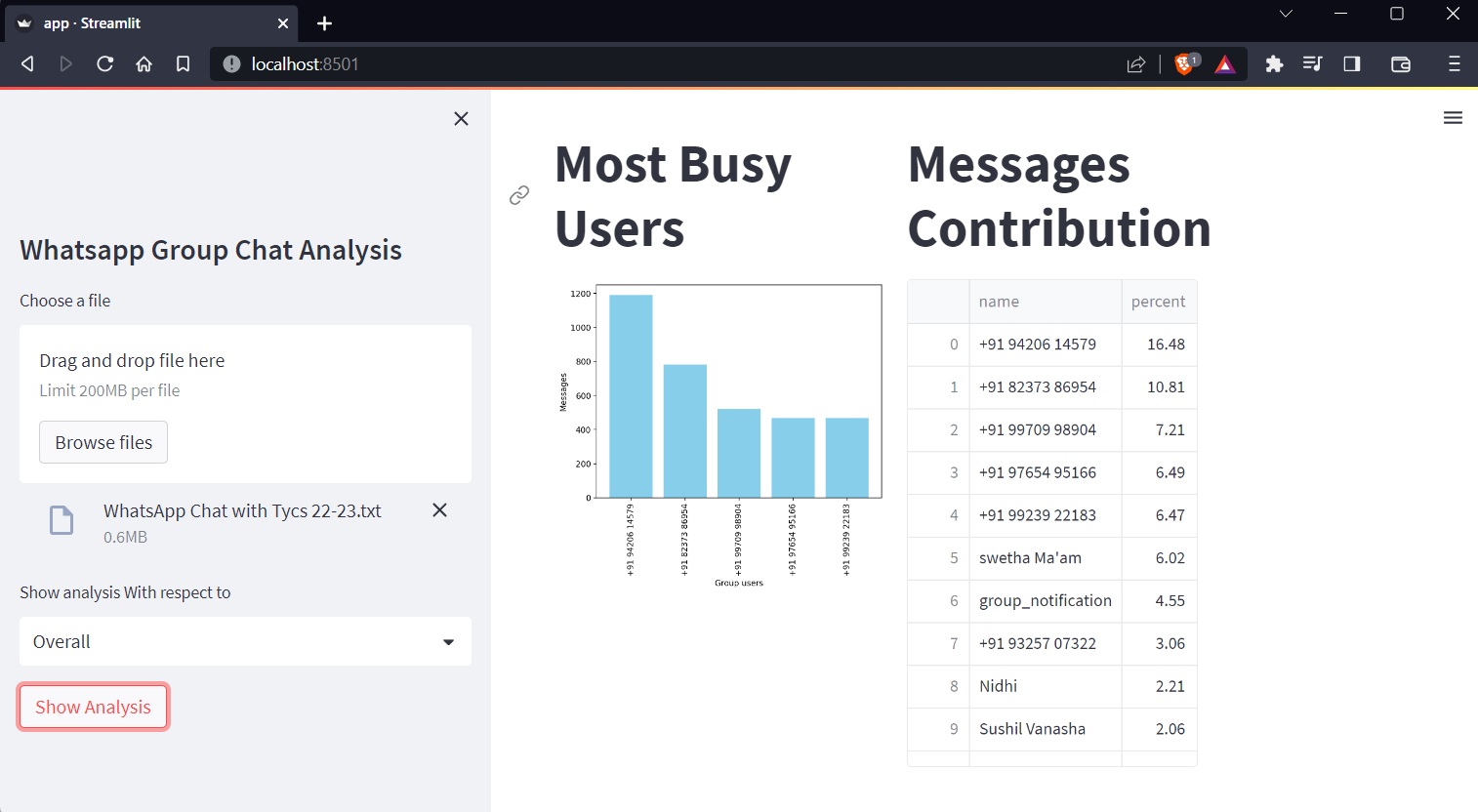
**Activity Map :**



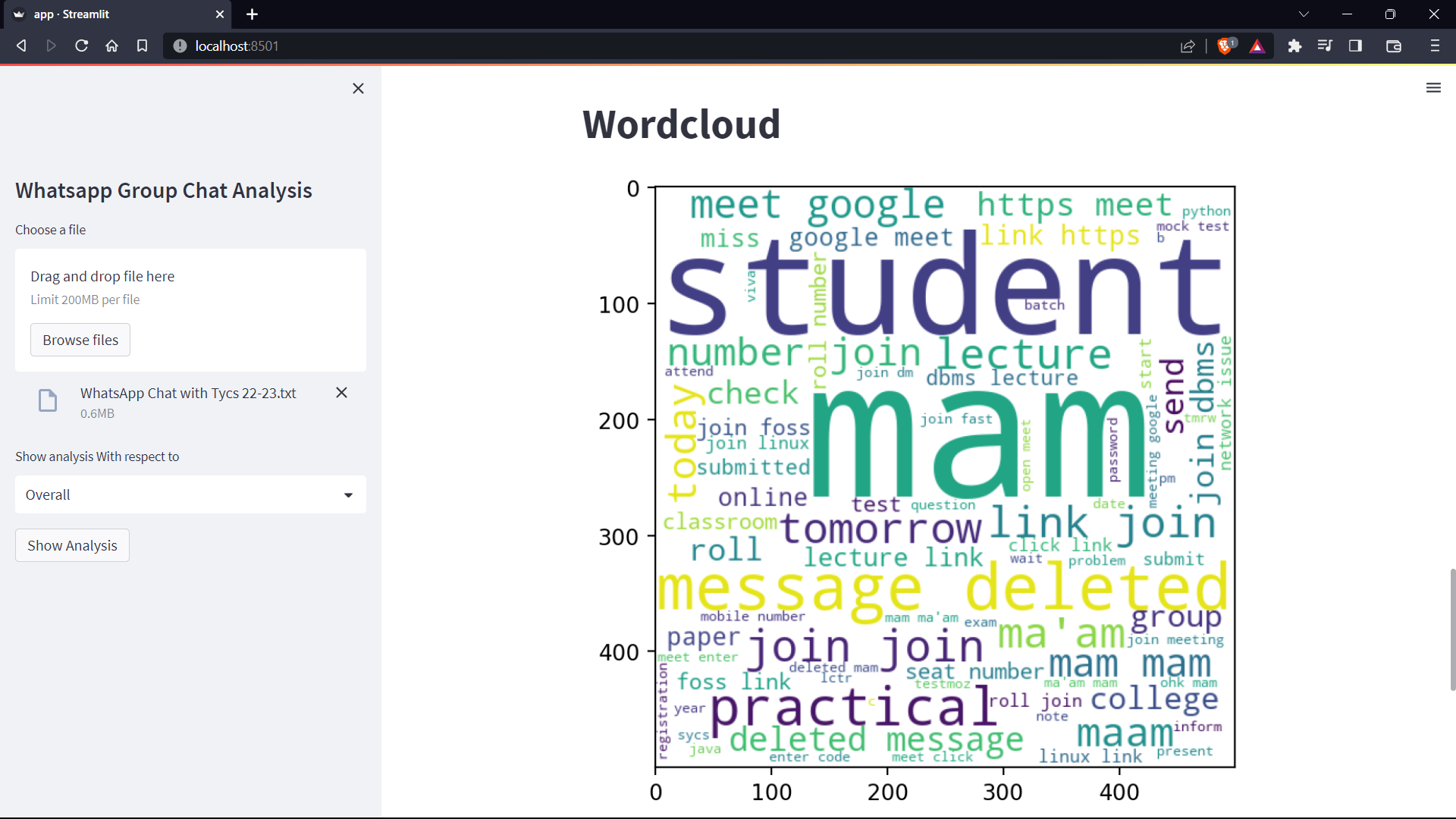
**Weekly Activity Map (Heatmap) :**

****

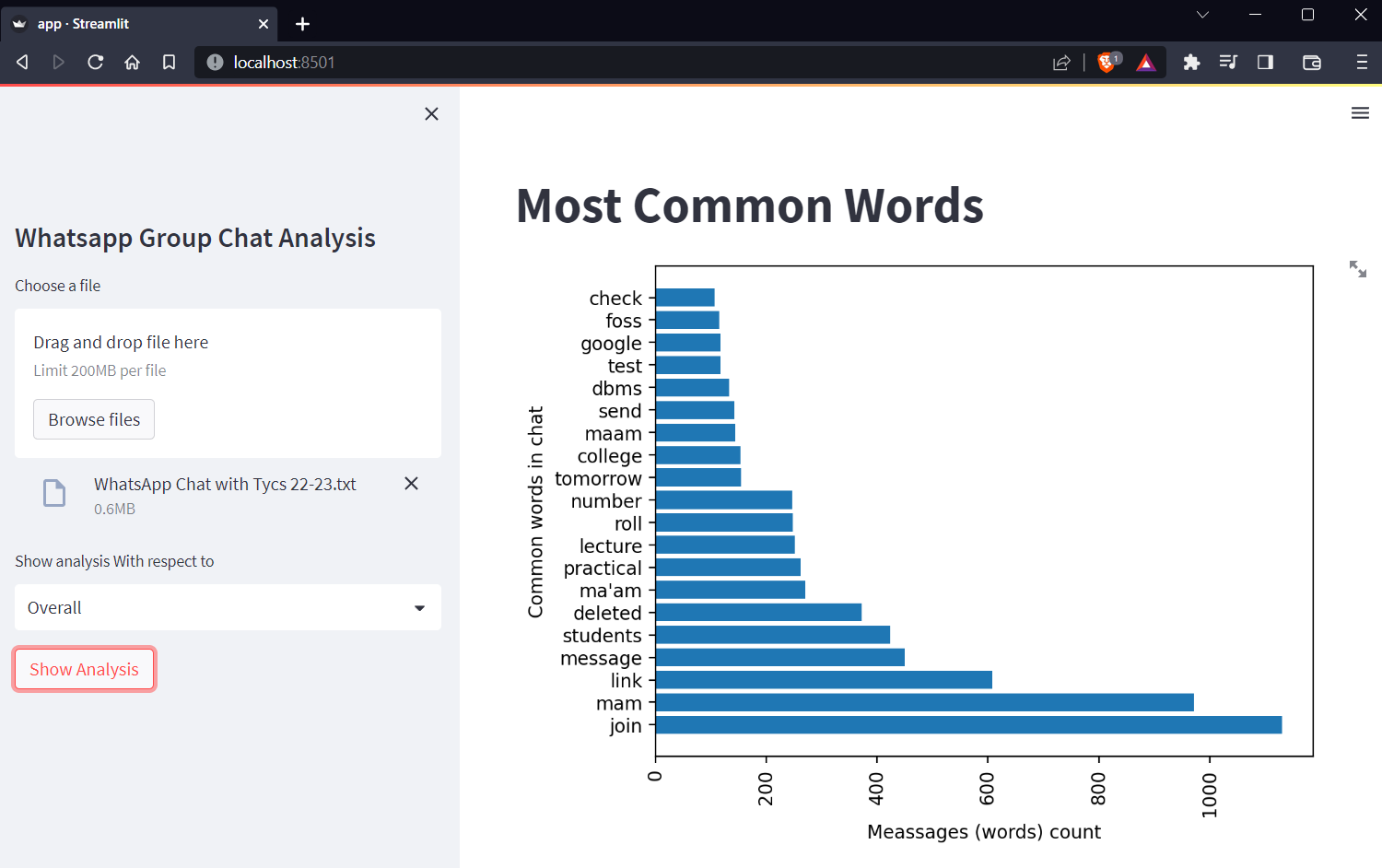
**Most Busy Users & Messages Contribution :**



**WordCloud :**



**Most Common Words :**

****

**CONCLUSION & FUTURE SCOPE**

**CONCLUSION**

This project is developed successfully and the performance is found to be satisfactory and This project is designed to meet the requirements.

WhatsApp group chat analysis can provide significant benefits to businesses, social media marketers, healthcare professionals, educational institutions, and government agencies by allowing them to monitor conversations and track trends. It has the potential to transform various industries and fields, and its demand is expected to increase in the future. With the help of this technology, organizations can improve their decision-making process, develop better products and services, and cater to the needs and preferences of their customers. In summary, WhatsApp group chat analysis is a powerful tool that can help businesses and organizations stay ahead of the curve and adapt to changing market dynamics.

**FUTURE SCOPE**

The future scope of WhatsApp group chat analysis is vast and diverse, as there are several potential applications for this technology in different fields. Here are some potential areas where WhatsApp group chat analysis could be useful:

**Business:** WhatsApp group chat analysis can be used by businesses to understand their customers' needs, preferences, and behaviors. By analyzing group chats, businesses can get valuable insights into what their customers are talking about, what they like and dislike, and what they expect from the business.

**Healthcare:** WhatsApp group chat analysis can be used in healthcare to monitor patient conversations and identify health issues that require attention. This technology can also be used to identify health trends and patterns and develop better treatments and preventive measures.

**Education:** Educational institutions can use WhatsApp group chat analysis to monitor student conversations and identify areas where students need help. This technology can also be used to track student progress, identify learning gaps, and develop personalized learning plans.

**Government:** Government agencies can use WhatsApp group chat analysis to monitor conversations related to public policy, identify public sentiment, and respond to emerging issues in real-time.

In summary, the future scope of WhatsApp group chat analysis is vast and diverse, and it has the potential to transform several industries and fields. With the increasing popularity of WhatsApp and other messaging apps, the demand for WhatsApp group chat analysis is only expected to grow in the coming years.

**REFERENCES**

https://www.analyticsvidhya.com/blog/

Seaborn : https://seaborn.pydata.org/

Streamlit Documentation : https://docs.streamlit.io/