WHATSAPP CHAT ANALYZER

A REPORT

Submitted by

DINKAR PANDEY - 191B099

HARSH MAROLIA – 191B115

YASH ROOPCHANDANI – 191B293

NIKHIL MISHRA – 191B314

Under the guidance of**:**

Dr. KUNJ BIHARI MEENA

MAY 2023

Submitted in partial fulfillment for the award of the degree Of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Department of Computer Science & Engineering

JAYPEE UNIVERSITY OF ENGINEERING &TECHNOLOGY,

A-B ROAD, RAGHOGARH, DT. GUNA - 473226, M.P., INDIA

DECLARATION

I hereby declare that the work reported in the B. Tech. project entitled as **“WhatsApp Chat Analyzer”** in partial fulfillment for the award of degree of, Bachelor of Technology submitted at Jaypee University of Engineering and Technology, Guna, as per best of my knowledge and belief there is no infringement of intellectual property right and copyright. In case of any violation, I will solely be responsible.

DINKAR PANDEY – 191B099

HARSH MAROLIA – 191B115

YASH ROOPCHANDANI – 191B293

NIKHIL MISHRA – 191B314

Department of Computer Science and Engineering,

Jaypee University of Engineering and Technology,

Guna, M.P., India

17th MAY 2023

JAYPEE UNIVERSITY OF ENGINEERING & TECHNOLOGY
(Accredited with Grade-A by NAAC & Approved U/S 2(f) of the UGC Act, 1956)
A.B. ROAD, RAGHOGARH, DIST: GUNA (M.P.) INDIA Phone:
07544 267051, 267310-14, Fax: 07544 267011 Website: www.juet.ac.in

CERTIFICATE

This is to certify that the work titled **"WhatsApp Chat Analyzer"**, submitted by "**DINKAR PANDEY (191B099), HARSH MAROLIA (191B115), YASH ROOPCHANDANI (191B293), NIKHIL MISHRA (191B314)"** in partial fulfilment for the award of degree of Bachelor of Technology (Computer science Engineering) of Jaypee University of engineering & Technology, Guna has been carried under my supervision. As per best of my knowledge and belief there is no infringement of intellectual property right and copyright. Also, this work has not been submitted partially or whole to any other university or Institute for the award of this or any other degree or diploma. In case of any violation concern student will solely be responsible

Signature of Supervisor

Dr. KUNJ BIHARI MEENA

17th MAY 2023

ACKNOWLEDGEMENT

Any endeavor cannot lead to success unless and until a proper platform is provided for the same. This is the reason we find ourselves very fortunate to have undergone our major project work under the supervision of **Dr. KUNJ BIHARI MEENA**. All the faculty of the department have extended a warm and helping hand.

Our sincere gratitude to **Dr. KUNJ BIHARI MEENA**, our major project supervisor for having faith in us and thus allowing us to carry out a project on a technology completely new to us. He helped immensely by guiding us throughout the course of the project.

Last but not least, I would like to acknowledge the contributions of all the resources, references, and materials that I have utilized during the course of this project. The knowledge and insights gained from these sources have been instrumental in expanding my understanding of web development and enhancing the quality of my work.

In conclusion, I extend my sincere appreciation to everyone who has contributed to the successful completion of my B.Tech web development project. Without their support, this endeavor would not have been possible. Their guidance, encouragement, and belief in me have been truly invaluable.

DINKAR PANDEY

HARSH MAROLIA

YASH ROOPCHANDANI

NIKHIL MISHRA

17th MAY 2023

EXECUTIVE SUMMARY

In this project we have proposed a WhatsApp Chat Analyzer. WhatsApp chats contain of different types of communications held among groups and personal chats. This chat contains of different topics. This can provide more data for technologies like machine learning. Machine learning models provides right learning experience which is important thing and indirectly affected by the data which provided to that model. This application provides analysis of this data which is WhatsApp provides. The advantage of this application is that it is implemented by simple python libraries like seaborn, pandas, numpy, streamlit and matplotlib which are commonly used for creating data frames and different graphs. This is displayed in web using heroku link which can run on all devices which supports browser.

The most used and efficient method of communication in recent times is an application called WhatsApp. WhatsApp chat analyzer is the application deployed on heroku web which provide analysis of WhatsApp group. There are various methodologies available for analysis but here matplotlib, streamlit, seaborn, re, pandas libraries of python and some concept of NLP is used. This is the combination of machine learning and NLP. This whatsapp chat analyzer imports whatsapp chat file from user and analyse it and give different visualizations as a result.

LIST OF FIGURES

Table of Contents

CHAPTER 1

INTRODUCTION

1. Introduction

The "WhatsApp Chat Analyzer" project aims to develop a software tool that offers valuable insights and analysis of WhatsApp chat conversations. With the widespread use of WhatsApp for personal and professional communication, understanding and extracting meaningful information from chat data has become increasingly important. The WhatsApp Chat Analyzer will provide users with a comprehensive understanding of their conversations, enabling them to gain insights, identify patterns, and improve communication dynamics. By processing chat exports, the application will generate statistics, word usage analysis, conversation trends, and more. Through an intuitive interface, users will be able to import chat files, visualize data, and generate reports, empowering them to enhance their communication strategies. This project seeks to bridge the gap between raw chat data and actionable insights, making WhatsApp conversations a valuable resource for personal and professional growth.

1.1. Problem Definition

"WhatsApp Chat Analyzer" is to address the challenges faced by individuals and organizations in extracting meaningful insights and information from WhatsApp chat conversations. Currently, there is a lack of efficient tools to analyze and visualize chat data, hindering the ability to identify patterns, trends, and key communication metrics. Without a dedicated chat analysis solution, users struggle to gain valuable insights from their WhatsApp chats, hindering their ability to improve communication dynamics, understand user behavior, or identify important information hidden within the conversations. The goal of this project is to develop a software tool that automates the analysis process, provides visualizations, and generates comprehensive reports to enable users to gain actionable insights from their WhatsApp chats efficiently and effectively.

1.2. Purpose

The most used and efficient method of communication in recent times is an application called WhatsApp. WhatsApp chats consists of various kinds of conversations held among group of people. This chat consists of various topics. This information can provide lots of data for latest technologies such as machine learning. The most important thing for a machine learning model is to provide the right learning experience which is indirectly affected by the data that we provide to the model. This tool aims to provide in depth analysis of this data which is provided by WhatsApp. Irrespective of whichever topic the conversation is based our developed code can be applied to obtain a better understanding of the data. The advantage of this tool is that is implemented using simple python modules such as pandas, matplotlib, seaborn and sentiment analysis which are used to create data frames and plot different graphs, where then it is displayed in the flutter application which is efficient and less resources consuming algorithm, therefore it can be easily applied to largest dataset.

1.3. Scope

The scope of the project "WhatsApp Chat Analyzer" is to develop a software tool that analyzes WhatsApp chat conversations and provides valuable insights and statistics. The application will be designed to process chat data from WhatsApp exports and extract information such as message frequency, word usage, emoji analysis, conversation trends, and participant activity. The software will offer a user-friendly interface for importing chat files, visualizing data, generating reports, and allowing users to customize their analysis preferences. The primary goal is to provide users with a comprehensive understanding of their WhatsApp conversations, enabling them to gain insights, identify patterns, and improve communication dynamics.

1.4. Hardware Specification

Processor- dual core 2.4 GHz (i5 or i7 series Intel processor or equivalent AMD)

RAM: 4 GB or above

Hard Disk: 1GB or above

Operating System Windows 10 or above

1.5. Software Specification

- **Python**

  Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. It supports multiple programming paradigms, including structured, object-oriented, and functional programming. It is commonly used for **developing websites and software, task automation, data analysis, and data visualization.**

- **PyCharm**

  PyCharm is an Integrated Development Environment used for programming in Python. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems, and supports web development with Django.

- **PIP**

  PIP is the standard package manager for Python. It allows you to install and manage additional packages that are not part of the Python standard library.

- **Jupyter Notebook**

  Jupyter Notebook allows users to compile all aspects of a data project in one place making it easier to show the entire process of a project to your intended audience. It is the original web application for creating and sharing computational documents. It offers a simple, streamlined, document centric experience.

CHAPTER 2

LITERATURE REVIEW

2. Literature Review

In this section, a brief introduction about the Current System i.e., the phases and processes which are involved during the development of a database and integrating it with an enterprise application and the Proposed System is given.

2.1. Current System

In olden days' there is no analysis for WhatsApp chat. If someone wants to analyze there is no CSV file to analyze. WhatsApp Application provide export txt file which is in raw format. It is very complicated for analysis. So, we have to forget that system and switch over to the WhatsApp Chat Analyzer.

Disadvantages of Existing System:

1. Raw data.
2. Time consuming.
3. Difficult to Analyze.
4. Analysis is not accurate

WhatsApp-Analyzer is a statistical analysis tool for WhatsApp chats. Working on the chat files that can be exported from WhatsApp it generates various plots showing, for example, which another participant a user responds to the most. We propose to employ dataset manipulation techniques to have a better understanding of WhatsApp chat present in our phones.

A survey of WhatsApp group analysis was done by [1], his work was focused on predicting the level of addiction of an individual to the WhatsApp group with respect to the age group and gender. He made use of R statistics software program.

The Research provided the basic idea of Statistical analysis. The work was done on a particular WhatsApp group data, to determine the type of communication medium people prefer the most in WhatsApp group chats, check most active day of week, to find which age group participants are more active. Also, to ascertain if the male users are more addicted to the WhatsApp group than females. Also, [2], in their work, presented a study that explores the classroom communication between teaching staff, faculty staff and high school students using WhatsApp. Their work focused mainly on getting a better understanding of the functions of the WhatsApp groups in the relationship between students and teachers, the activities they carry out with the application and the way it generally affects educational and academic output.

2.2. Proposed System

The “WhatsApp Chat Analyzer” provides a platform to the user which enables user to analyze WhatsApp chats. This application allows user to browse WhatsApp exported (.txt) file and import it to WhatsApp chat analyzer and get analysis according to that txt file. And user can Analyze by clicking Show Analysis button.

Advantages of WhatsApp Chat Analyzer:

- Runs on all devices.
- Shows based on WhatsApp chat file.
- Shows different visualizations.
- Total Messages.
- Total words.
- Media shared.
- Link shared.
- Monthly timeline.
- Most busy day

CHAPTER 3

TOOLS AND TECHNOLOGY

3. Tools and Technology

In this section, a brief introduction about the Current System i.e., the phases and processes which are involved during the development of a database and integrating it with an enterprise application and the Proposed System is given.

3.1. WhatsApp

The WhatsApp is a widely popular messaging application that has revolutionized the way people communicate in the digital age. It allows users to send text messages, make voice and video calls, share multimedia content, and create group chats. With its user-friendly interface and extensive global user base, WhatsApp has become a primary mode of communication for billions of people worldwide. This project aims to delve into the realm of WhatsApp conversations and analyze the patterns and behaviors exhibited by its users. By examining the vast amount of chat data available, this study seeks to uncover valuable insights about how individuals interact, the topics they discuss, the frequency of their messages, and even their emotional states. The findings from this analysis will contribute to a deeper understanding of human communication in the digital era and can have implications for various fields such as social sciences, psychology, and data analytics.

3.2. Pandas

This is an open-source Python libraries which is mainly used in Data Science and machine learning subjects. This library provides analysis tool for data manipulation, using its data structures this are used for analyzing data for manipulating time series analysis and numerical data.

3.3. NumPy

NumPy can be name come from Numeric Python, it is a data analysis library for Python that contains various numerical functions and methods for numerical analysis and also having multi-dimensional array objects and to process these arrays contains collection of routines.

3.4. Matplotlib

Matplotlib is easy to use and an amazing visualizing library in Python. It is built on NumPy arrays and it work with the broader SciPy stack and consists of several plots like pie, line, bar, graph, scatter, histogram, etc. In this project, Matplotlib is used for various visualizations for analysis of WhatsApp chats. Visualizations like bar charts, line charts, pie charts are used.

3.5. Seaborn

Seaborn is a library mostly used for statistical plotting in Python. To make statistical plots more attractive it provides beautiful color palettes and default styles. In this project, Seaborn is used for heatmap visualization for showing 24 hours with 7 days with different scale of color for getting hour with max to min messages.

3.6. Streamlit

In this project, this library is used for creating beautiful web items and objects for representing WhatsApp chat analysis with different types of charts and visualizations on Streamlit

3.7. NLP

In this project, Features of NLP are used like Parsing Text, eliminating stop words and Analyzing Text. Parsing text is used for splitting messages into words for analysis like total words and mostly used words. A file is used that contains all stop words which is given to the python program to show meaningful words only by eliminating all stop words. Text analysis is used to identify how many media are shared, how many links are shared.

3.8. Deployment

Deployment means pushing changes or updates from one deployment environment to another. When setting up a website we will always have our live website, which is called the live environment or production environment. If we want the ability to make changes without these affecting our live website, then we can add additional environments. These environments are called development environments or deployment environments. The additional development environments will typically be a local environment, a development environment, and a staging environment. How many environments we need is up to we and depends on the complexity of the project we are working on. While deployment models can vary, the most common is the classic "left to right" deployment model when working with multiple deployment environments. In this model, changes are made in local, development, or staging environments (depending on the setup) and pushed from left to right through the different environments ending up in the live environment. Once this deployment process has been completed the new changes will be visible in the live environment.

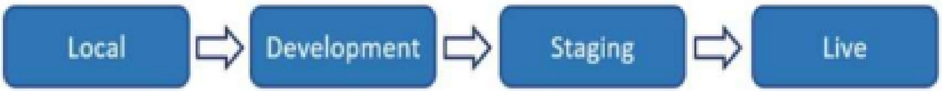


Fig. 3.1. Process of Deployment

The above image shows a very simplified and classic way of handling deployments when working with websites in a CMS. We do not necessarily need all of the above environments, but the process stays the same. By using multiple environments, we get a list of advantages - the main one being, that we can make changes without it affecting our live website. Once the changes are made, tested, and ready to be pushed live, the deployment process takes care of the rest.

The deployment process flow consists of 5 steps:

- Planning
- Development
- Testing
- Deploying
- Monitoring

3.9. Git

This one might seem obvious, but having a version control system is invaluable to any good deployment workflow. Without it, it is likely that mistakes will happen if we are working in a team. Even if we are the only developer working on a project it's highly recommended to use Git in case we need to go back to previous versions or if someone new joins our team. Without Git it will be hard to ensure consistency in our deployment workflow and can lead to more mistakes being made from deploying unfinished code or not having all team members work on the same version of the code. While it is possible to work directly on a development environment, it will in most cases save we a lot of time by working locally instead. By installing the website or software locally we will be able to work more efficiently while also speeding up testing and verification of our code.

The only downside of running locally is that we must install the website or software on our machine. But the time spent doing so is won back multiple times in the long run. Firstly, we do not have to constantly commit, push, and deploy a change before we can verify if it works. And when something does not work, we will have to revert it, push it again and redeploy.

Instead, we can simply run it all locally and once it is working as intended, we can push it directly to the staging environment for more rigorous testing.

CHAPTER 4

RESEARCH METHODOLOGY

4. Research Methodology

This section contains a description of the research methodologies followed during the implementation of the project. The project was implemented with Agile methodology. Brief descriptions of the methodologies are given below:

4.1. Waterfall Methodology

The waterfall model is the first modern approach to the (SDLC) software development life cycle model. The model describes the project development in multiple sequential phases. Each phase track progress of the project from multiple dimensions and the result of each phase act as input for the next phase. The description of the sequential phases that comprise the waterfall model are as follow:

- **Requirement Analysis:** In the first phase of the waterfall model, all the business requirements and logic of the system are documented. The requirements are divided into two categories viz Functional requirements and Non-Functional requirements. Functional requirements define the system behavior while Non-functional requirements define how the system should behave. These requirements are gathered through the discussion of the end-users and/or clients and finally validated for the possibility of the implementation with given time and resources. The documented requirement serves as a guideline for the next phase.

- **System Design:** In the second phase, all the collected requirements are taken into consideration to prepare documentation for the system and software design. In this phase, there are two types of design development High-Level Design (HLD) and Low-Level Design (LLD). HLD describes the properties of every module such as description, name, outline, functionality, relationship, identification of database table, and complete architecture diagram along with technical details. And LLD is more of a description of the functional logic of the modules, database table with the properties like type and size, complete interface details, handling errors, and input/output for every module.

- **Implementation:** In the third phase, the coding of the application begins according to the specification of the system design. The system is built using the chosen programming of language/s. The tasks are divided into multiple smaller units which are assigned to the developers. Along the way, the unit or the modules are integrated into larger functional components of the system as described in the previous phases. This is normally the longest-running phase-out of all the phases in the waterfall model.
- **Testing:** Testing is the fourth phase of the waterfall model. Once, the system is ready, it is deployed or hosted in a testing environment. The testing of the system is carried out to verify functional and non-functional requirements are met which is set by the customer during the requirement analysis phase. During the testing, the possible bugs and defects are tracked and reported back to the developers, and fixed. The system is tested multiple times and the process continues until the application is stable, bug-free as much as possible, and covers all the business requirements.
- **Deployment:** In this phase, the application is deployed or packaged for installation to the environment where the end-users can start using it. The system is monitored for any possible deployment issues.
- **Maintenance:** After the deployment or release of the system, the end-users will start using it. During this phase, three major activities are performed Bug fixing, upgrade, and enhancement. Feedbacks are taken from the end user's experience and any bugs that are tracked will be fixed. It ensures the system is consistently performing as per the specification.

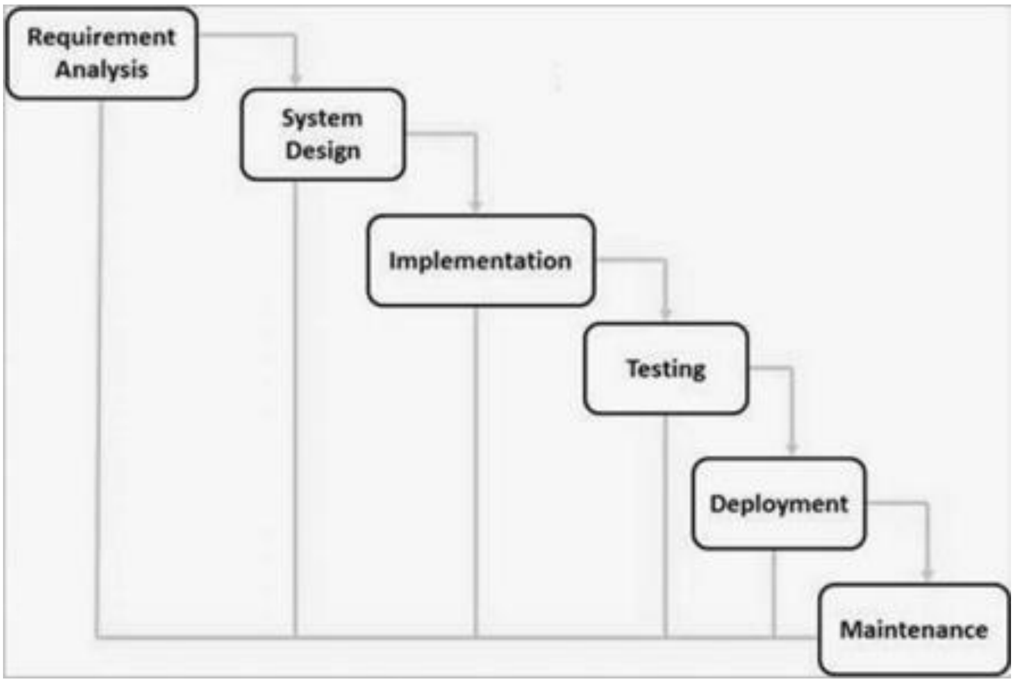


Figure 4.1 – Waterfall Model

4.2. Agile Methodology

Agile methodology is one of the project management methodologies, it breaks down the development into multiple iterative cycles. Each cycle lasts one to three weeks and involves team collaboration across multiple dimensions of project development such as Planning, Requirement Analysis, Design, Implementation, Testing, and Delivery continuously. This approach allows the project to be adjusted and remodeled easily and reach the goal with updated requirements that fits the latest business need.

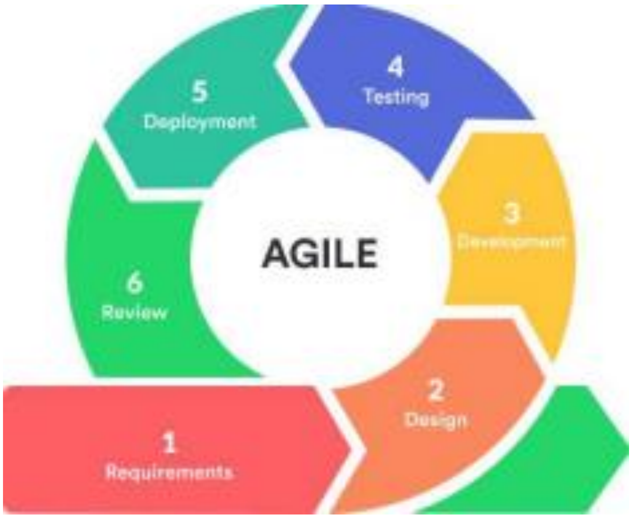


Figure 4.2 – Agile Software Development Cycle

4.3. Hybrid Methodology

The goal of a hybrid method is to be able to both define the requirements up front (planning, budgeting, etc.) as with a waterfall approach, and then to move onto an agile approach when it comes to design, development, and testing. Hybrid combines the best of agile and waterfall and makes them work together. This provides benefits of both methodologies and to iterate the logic, design, and features of the application much conveniently.

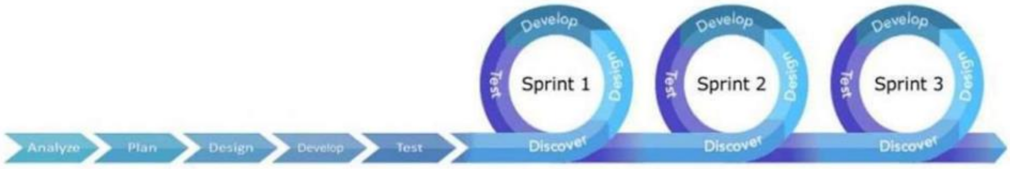


Figure 4.3 – Hybrid Model

CHAPTER 5

FEASIBILITY STUDY

5. Feasibility Study

The feasibility study is the following step after the problem has been clearly defined and solutions have been proposed. A feasibility study is an assessment or analysis of a proposed project's or program's possible impact. The system's needs are defined by a collection of constraints, such as system objectives and output descriptions. The analyst must then assess the viability of the proposed system in producing the above results. During the feasibility research, three major elements must be examined. The goal is to see if the suggested system can be implemented. The suggested system is exposed to three parts of the feasibility study, which are explained below.

5.1. Economic Feasibility

Economic feasibility is the most frequently used method for evaluating the effectiveness of the new system. Without a doubt, this measure is most often and important one of the three. Information systems are often viewed as capital investments for the business, and as such should be subjected to the same type of investment analysis as other capital investments. Economic analysis is used for evaluating the effectiveness of the proposed system. In economic feasibility, the most important is cost-benefit analysis. This project is not economical as it mainly depends on the sharing of data between two phones.

5.2. Technical Feasibility

It is the measure of the specific technical solution and the availability of the technical resources and expertise. It is one of the first studies that must be conducted after tool has been identified. A technical study of feasibility is an assessment of the logistical aspects of business operation. This is considered with specifying equipment and software that will successfully satisfy the user requirement. The technical needs of the system may vary considerably but should include the facility to produce outputs in a given time, response time under certain conditions and the ability to process a certain amount of transaction at a certain speed. The proposed system is developed by using Jupyter software. Jupyter is non-profit organization created to develop open-source software, open standards, and services for interactive computing across dozens of programming languages. The idea is to implement a data processing code using python to make better sense of WhatsApp group chat data.

5.3. Operational Feasibility

Operational feasibility is mainly concerned with issues like whether the system will be used if it is developed and implemented, whether there will be resistance from the users which will affect the possible application benefits. It is the ability to utilize, support and perform the necessary tasks of a system or program. It is the measure of how well a proposed system solves the problem and takes advantages of the opportunities identified during the scope definition and problem analysis phases. This system helps in many ways. It shows the number of users using WhatsApp and gives the data information of their sharing data. Which is organized in Pie-chart and Bar-chart.

CHAPTER 6

SYSTEM DESIGN

6. System Design

System Design is the process of designing the architecture, components, and interfaces for a system so that it meets the end-user requirements.

6.1. Process Flow Diagram

Flowcharts are nothing but the graphical representation of the data or the algorithm for a better understanding of the code visually. It displays step-by-step solutions to a problem, algorithm, or process. It is a pictorial way of representing steps that are preferred by most beginner-level programmers to understand algorithms of computer science, thus it contributes to troubleshooting the issues in the algorithm. A flowchart is a picture of boxes that indicates the process flow in a sequential manner. Since a flowchart is a pictorial representation of a process or algorithm, it is easy to interpret and understand the process.

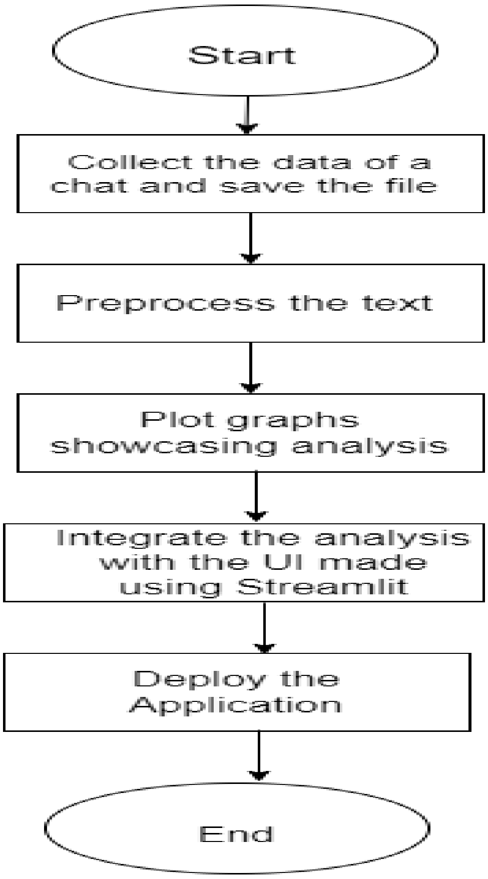

Start
Collect the data of a chat and save the file
Preprocess the text
Plot graphs showcasing analysis
Integrate the analysis with the UI made using Streamlit
Deploy the Application
End

Figure 6.1 – Process Flow Diagram of WhatsApp Chat Analyzer

CHAPTER 7

IMPLEMMENTATION

7. Implementation

7.1. Algorithm

The first step of this process is to obtain the conversation on TXT format. The easiest way to export an entire chat history, not including video or photos from your cell phone, is to use the built-in “Export Chat” feature, following these steps.

1. Open an individual or a group chat.
2. Tap the Menu button shown in the image below.
3. Tap the More button.
4. Select Export Chat.
5. Tap Without Media from the options that are given.
6. Select an option to share TXT file.

Importing Libraries

```
import re
import pandas as pd
from urlextract import URLExtract
from wordcloud import WordCloud
import pandas as pd
from collections import Counter
import emoji
import streamlit as st
import preprocessor,helper
import matplotlib.pyplot as plt
import seaborn as sns
```

Pre-processing the data in order to perform analysis on it. Adjusting and adding new columns.

Using another pandas method creating DataFrames, such as format, type of message and the count of records per column, in this way we found that format of DateTime column is string, we will transform this column to a suitable date and time format to be able to analyze these aspects.

```
def preprocess(data):
    pattern = '\d{1,2}/\d{1,2}/\d{2,4},\s\d{1,2}:\d{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='%m/%d/%y, %H:%M - ')
    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:]:
            users.append(entry[1])
            messages.append(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
```

Word Count

If want to know which day of the week of each month had the greatest number of messages sent, to obtain these data

1. Group data by month and day of the week, in addition to counting the messages sent.
2. Make a pivot_table with DataFrame obtained from in the previous step, having as columns days of the week and as rows the months of the year and the count carried out as the values to evaluate.
3. Using Seaborn, perform a HeatMap with the activity_heatmap function.

It’s possible to know the most used words in chat, following this steps:

1. Using stopwords, to remove some words from our data.
2. Generate a new DataFrame copying chat DataFrame selecting author and message columns.

3. Separate each word of each message to make a row with each of them.
4. Use "remove_emoji" function, to not consider emojis as common words.
5. Remove empty or NaN rows
6. Unifying every laughing text just in a "jaja"
7. Grouping most common words then count their repetitions
8. Using matplotlib make a nice chart with top 10 common words

WordCloud

It’s time to create a word cloud with all “chat” words, for this we will use WordCloud library. Using a couple of new functions to plot our chart and to eliminate links to web sites that could be found in messages.

Emojis

What data can we get from emojis in group chat? A sum of all different used emojis.

```
def emoji_helper(selected_user,df):
    if selected_user != 'Overall':
        df = df[df['user'] == selected_user]
    emojis = []
    for message in df['message']:
        emojis.extend([c for c in message if c in emoji.EMOJI_DATA])
    emoji_df = pd.DataFrame(Counter(emojis).most_common(len(Counter(emojis))))
    return emoji_df
```

7.2. Challenges

The development of a WhatsApp Chat Analyzer project presents several challenges that need to be addressed for successful implementation. Firstly, acquiring a substantial and diverse dataset of WhatsApp chats can be a challenge due to privacy concerns and legal limitations. Obtaining consent from participants and ensuring data anonymization are crucial aspects to navigate. Additionally, preprocessing the chat data poses its own set of challenges, including handling multimedia content, removing irrelevant or sensitive information, and addressing language variations and slang. Another significant challenge lies in extracting meaningful features from the chat data, such as topic modeling, and user behavior patterns, requiring advanced natural language processing techniques and algorithms. The analysis and interpretation phase may also encounter difficulties in identifying and interpreting complex conversational patterns and user behaviors accurately. Finally, ensuring the ethical use of chat data, respecting user privacy, and maintaining data security are ongoing challenges that need careful consideration throughout the project. Overcoming these challenges will contribute to the development of a robust and reliable WhatsApp Chat Analyzer.

CHAPTER 8

RESULTS

8. Results

The results obtained from the WhatsApp Chat Analyzer project provide valuable insights into conversational patterns and user behavior within the platform. Through extensive data collection and preprocessing, the project successfully extracted relevant features from the chat data, including message frequency, sentiment analysis, and topic modeling. The analysis and interpretation of these features revealed intriguing findings, such as the identification of dominant conversation topics, fluctuations in user sentiment over time, and distinctive communication patterns among different user groups. Additionally, user behavior modeling techniques allowed for the prediction and classification of user interactions, providing a deeper understanding of how individuals engage and communicate within WhatsApp. These results not only contribute to the field of chat analysis but also have practical implications in various domains, such as marketing, social sciences, and human-computer interaction.

Working of project:

1. User go to sidebar and click on browse file.
2. Select WhatsApp chat text file and import it for analysis.
3. User have choice for overall analysis or specific user analysis from whole group.
4. After selecting user, User click on show analysis button to analyze imported file.
5. It shows analysis of imported WhatsApp text file.
6. User can see Total messages, words, media, and link shared in the group.
7. Then monthly and daily timeline for the message is shown using line charts.
8. Activity Map in which most busy month and day is shown by the bar charts.
9. Then weekly activity map which shows hourly activity of users with corresponding day using heat map.
10. Top five busy users in group using graph and list of users with percentage of use.
11. WordCloud shows an interesting visualization of most common words.
12. Top ten most common word represented by using bar chart.
13. List of Emojis with number of times it is used.
14. Pie chart which shows top five emojis percentage of use.

This is the result of project and how project is working.

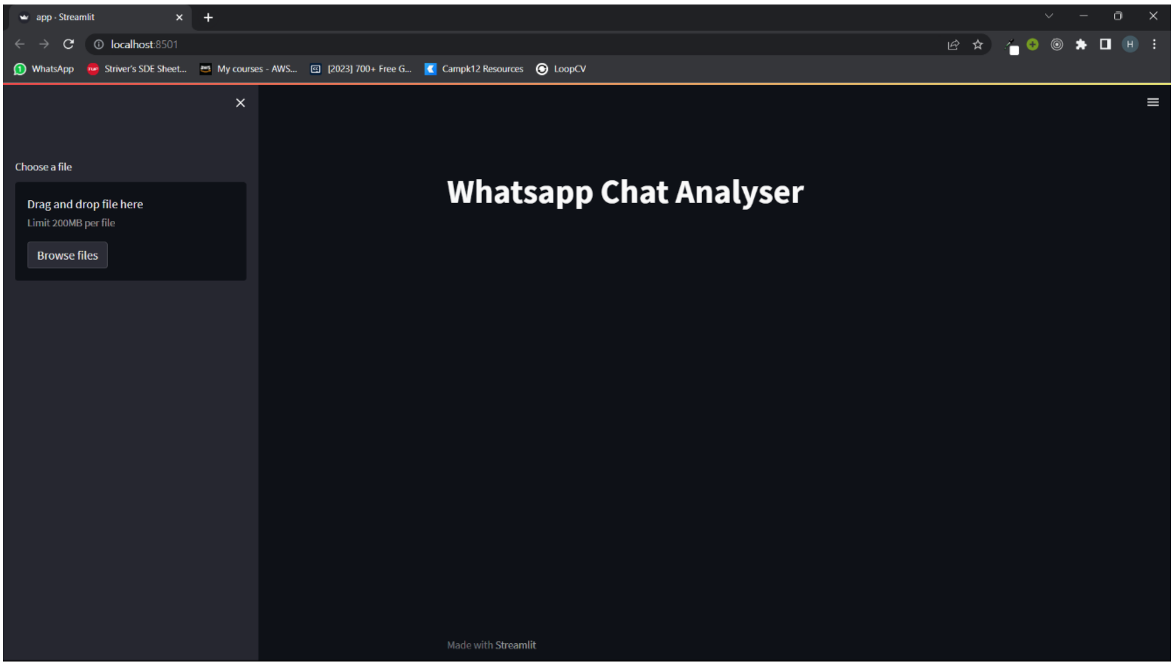


Figure 8.1 – Home Screen

The homepage of the WhatsApp Chat Analyzer project features an intuitive and user-friendly interface, displaying insightful visualizations and analytics of chat conversations.

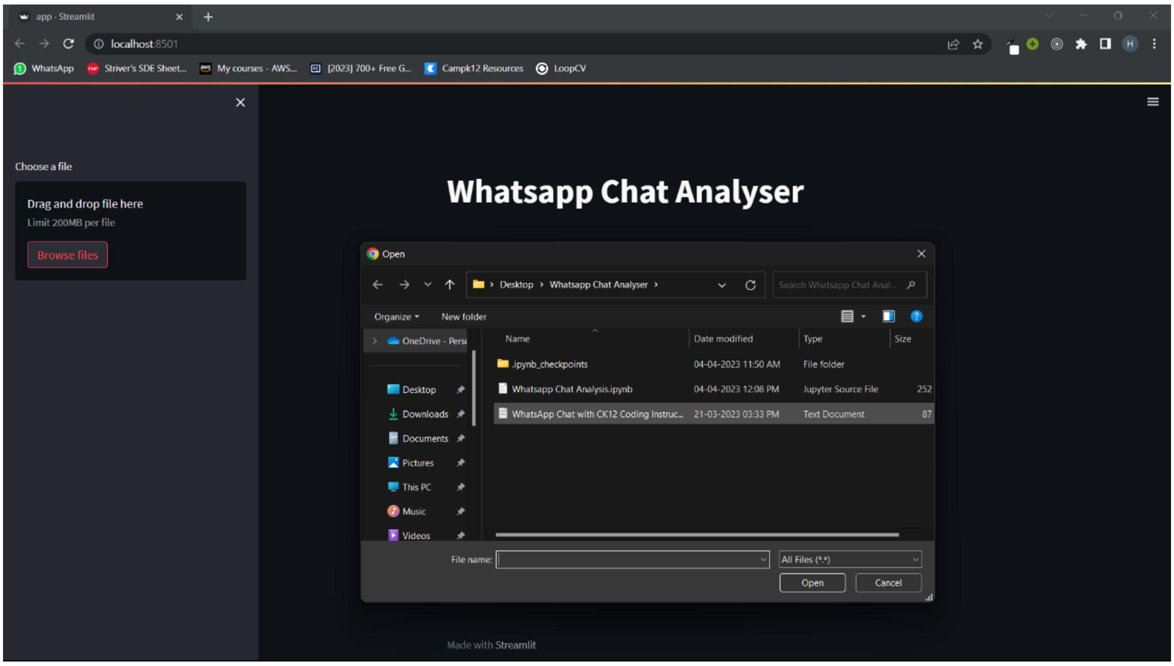


Figure 8.2 – File Upload

The image of the file upload page showcases a user-friendly interface allowing seamless and efficient uploading of WhatsApp chat data for analysis.

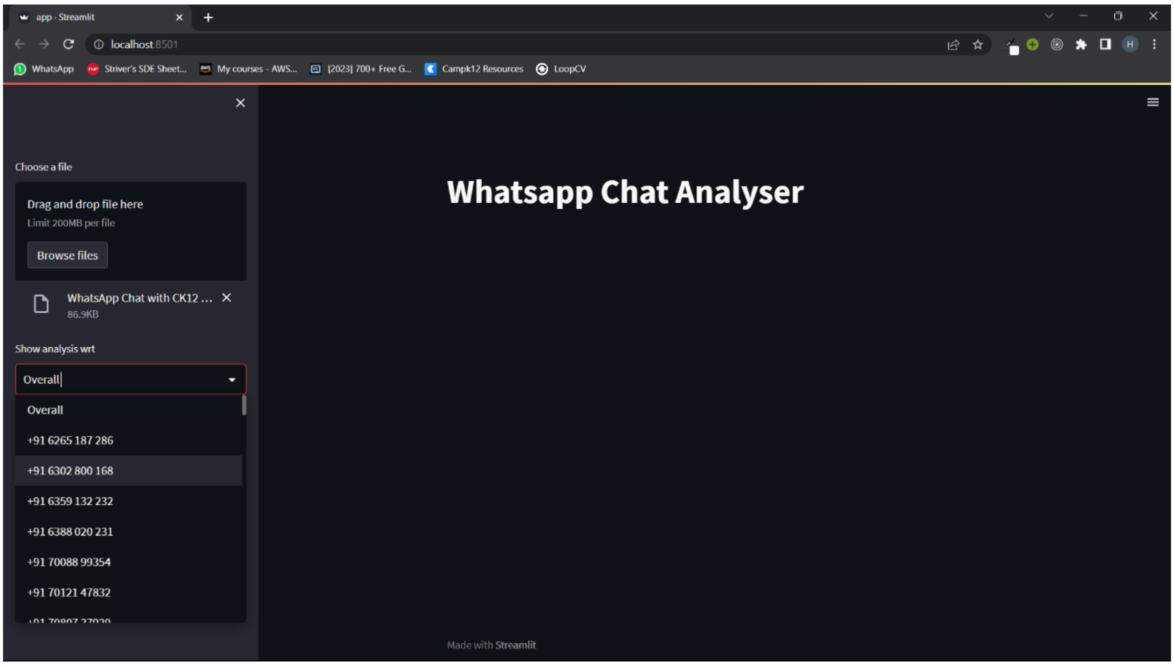
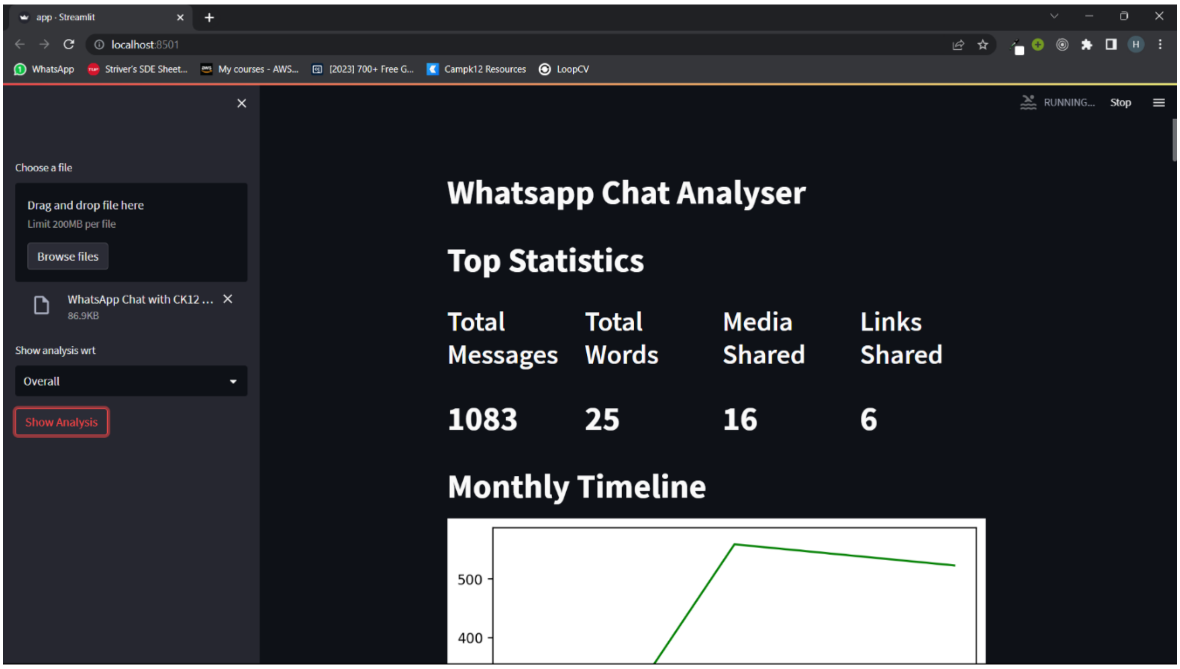


Figure 8.3 – Select User

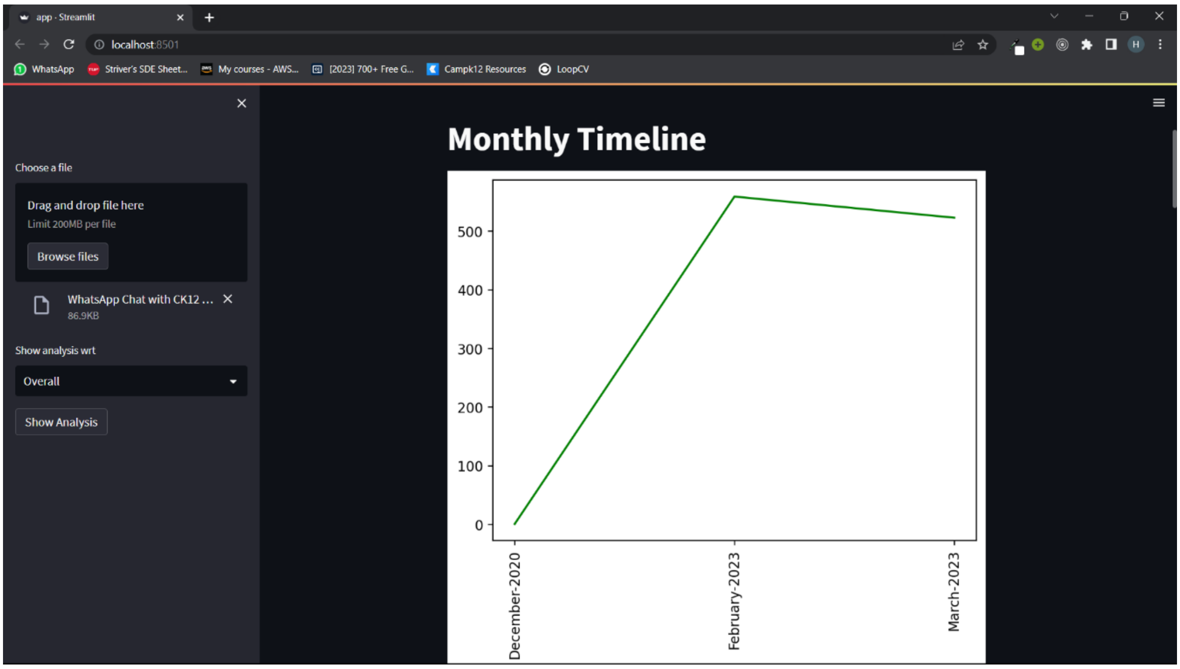
The image of the Select User page in the WhatsApp Chat Analyzer presents a user-friendly interface with a comprehensive list of chat participants, allowing users to choose specific individuals for in-depth analysis.



Total Messages	Total Words	Media Shared	Links Shared
1083	25	16	6

Figure 8.4 – Top Statistics

The Top Statistics page provides a comprehensive visual representation of key metrics and trends extracted from WhatsApp chat data.

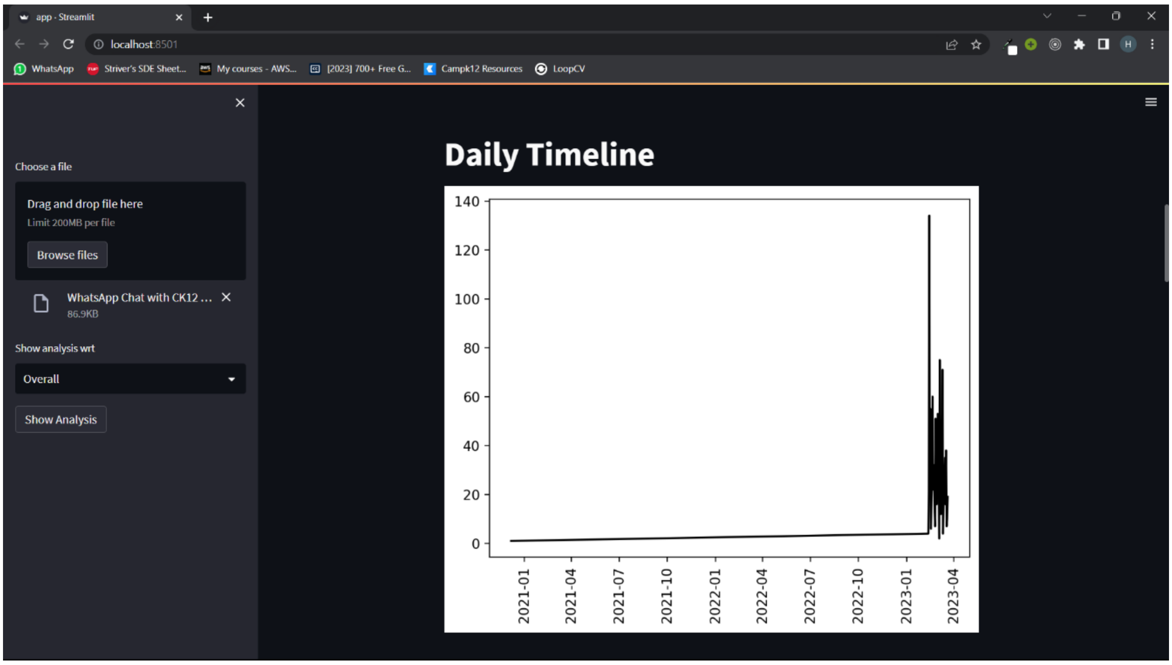


WhatsApp Chat with CK12 ... 86.9KB

Show analysis wrt: Overall

Figure 8.5 – Monthly Timeline

The Monthly Timeline page provides a visual representation of the chat activity over time, highlighting the fluctuations and key events throughout the analyzed period.

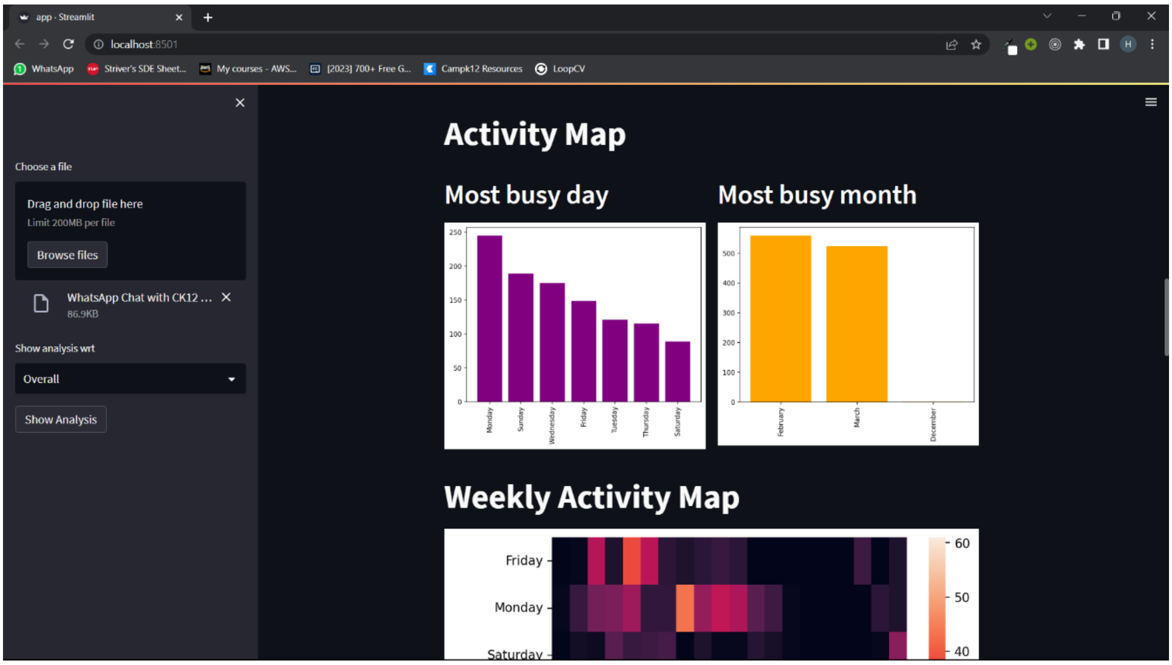


WhatsApp Chat with CK12 ... 86.9KB

Show analysis wrt: Overall

Figure 8.6 – Daily Timeline

The Daily Timeline page displays an interactive graphical representation of the daily activity patterns in the WhatsApp chat conversations.


Choose a file

Drag and drop file here
Limit 200MB per file

WhatsApp Chat with CK12 … 86.9KB

Show analysis wrt: Overall

Activity Map

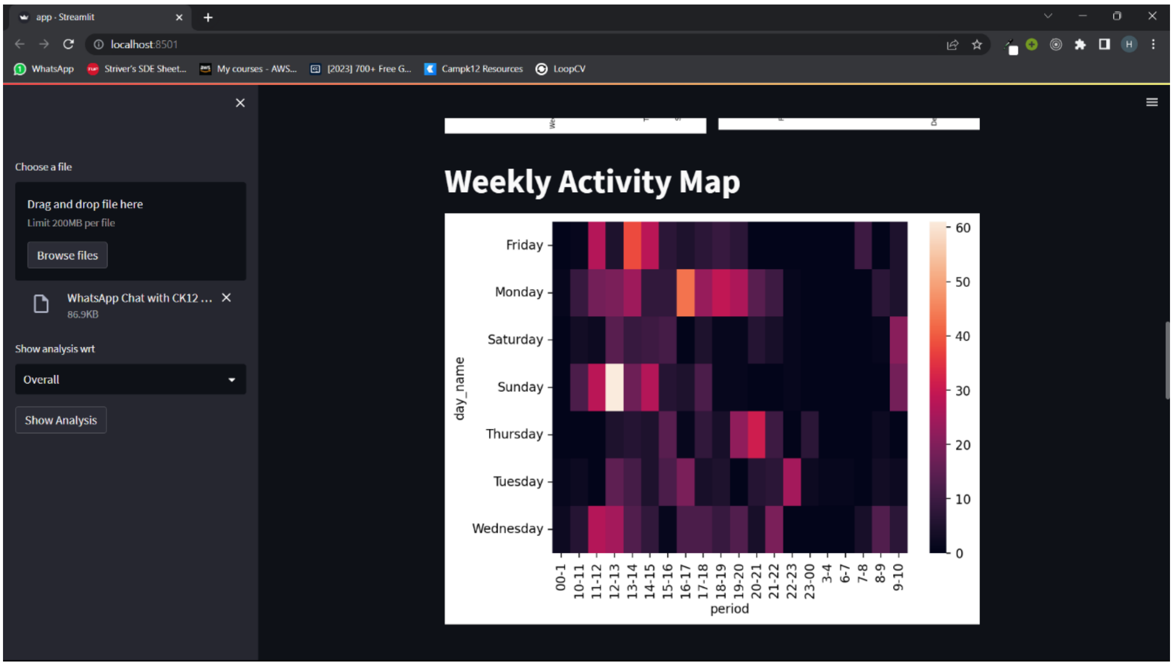
Most busy day

Most busy month

Weekly Activity Map

Figure 8.7 – Activity Map

The Activity Map page provides a visual representation of the frequency and intensity of chat interactions, offering an insightful overview of user engagement and group dynamics.


Choose a file

Drag and drop file here
Limit 200MB per file

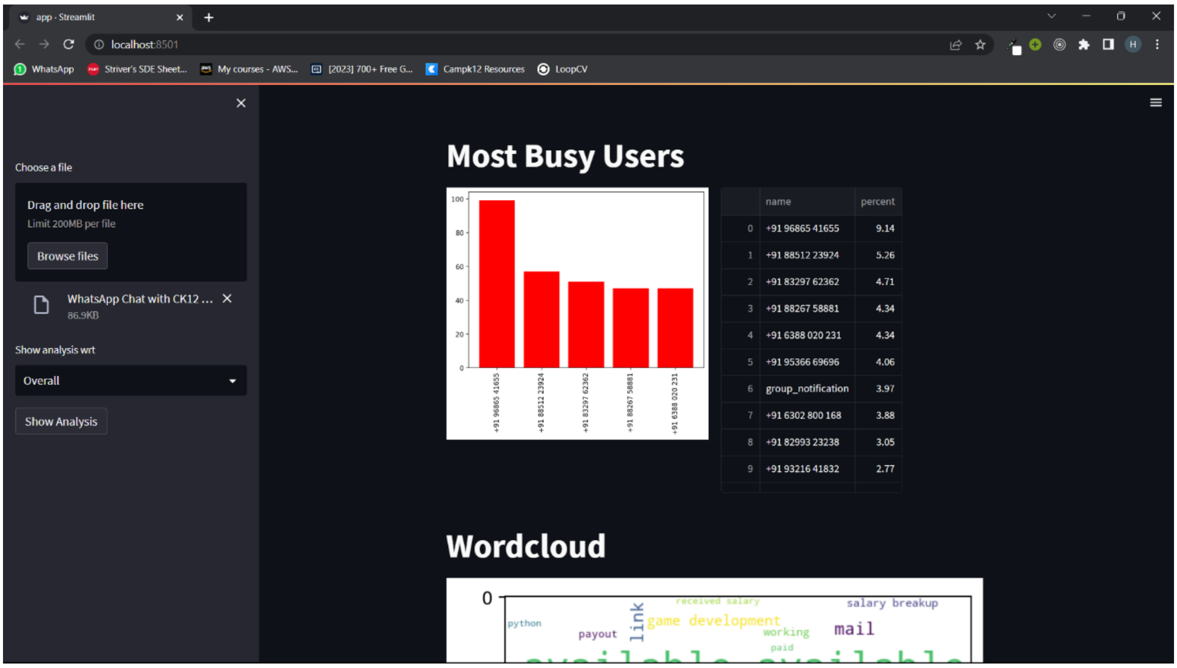
WhatsApp Chat with CK12 … 86.9KB

Show analysis wrt: Overall

Weekly Activity Map

Figure 8.8 – Weekly Activity Map

The weekly Activity Map page provides an intuitive visual representation of the chat activity patterns and engagement levels throughout the week.



WhatsApp Chat with CK12 ... 86.9KB

Show analysis wrt: Overall

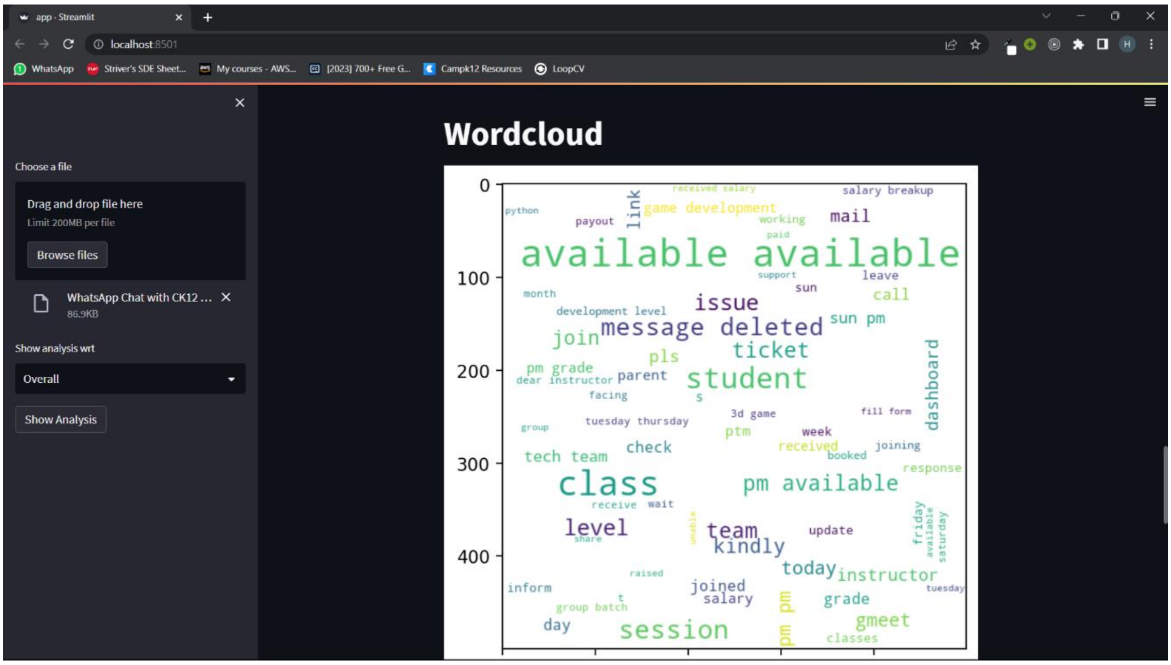
Most Busy Users

	name	percent
0	+91 96865 41655	9.14
1	+91 88512 23924	5.26
2	+91 83297 62362	4.71
3	+91 88267 58881	4.34
4	+91 6388 020 231	4.34
5	+91 95366 69696	4.06
6	group_notification	3.97
7	+91 6302 800 168	3.88
8	+91 82993 23238	3.05
9	+91 93216 41832	2.77

Wordcloud

Figure 8.9 – Most Busy Users

The "Most Busy Users" page displays a visual representation of the top users who have been most active in the WhatsApp chat conversations.



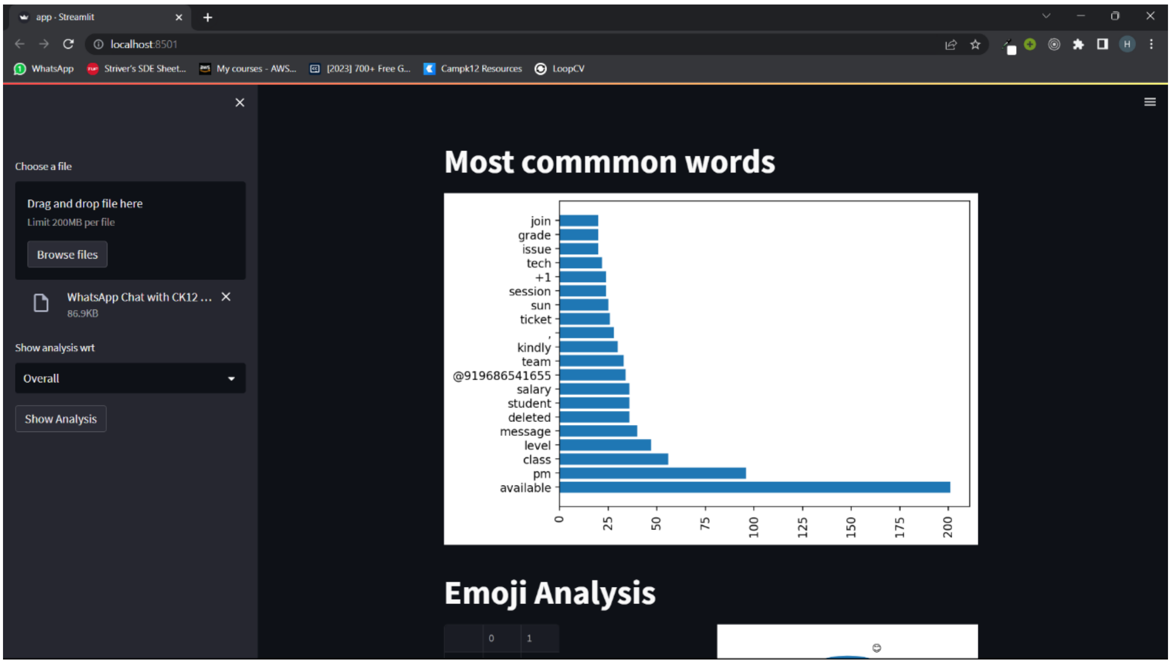
WhatsApp Chat with CK12 ... 86.9KB

Show analysis wrt: Overall

Wordcloud

Figure 8.10 – Wordcloud

The WordCloud visualization provides a captivating representation of the most frequently used words in the WhatsApp chat conversations.



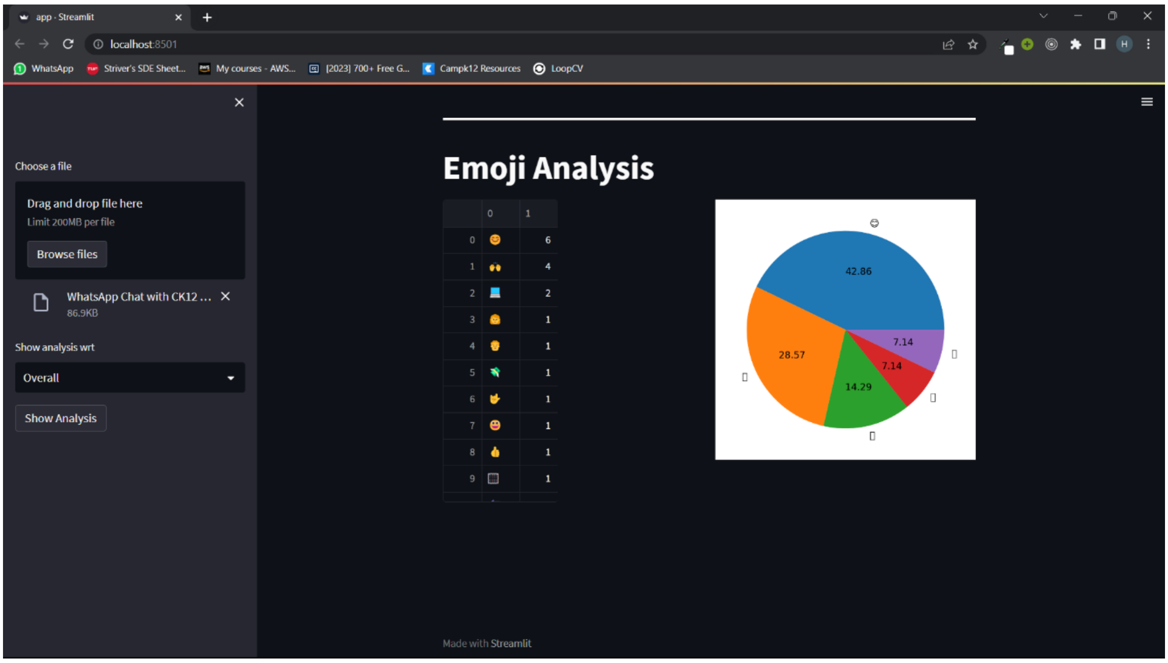
Choose a file

WhatsApp Chat with CK12 ... 86.9KB

Show analysis wrt: Overall

Figure 8.11 – Most Common Words

The Most Common Words page provides a visual representation of the frequently used words in WhatsApp chats, highlighting the key terms that dominate the conversations.



Choose a file

WhatsApp Chat with CK12 ... 86.9KB

Show analysis wrt: Overall

Emoji Analysis

	0	1
0	😊	6
1	🙌	4
2	💻	2
3	👩	1
4	👦	1
5	💸	1
6	🤟	1
7	😃	1
8	👍	1
9	▦	1

Figure 8.12 – Emoji Analysis

The Emoji Analysis page provides visual representations and insights into the usage and sentiment of emojis in WhatsApp chats.

CHAPTER 9

CONCLUSION AND FUTURE WORK

9. Conclusion and future work

9.1. Conclusion

Further The major objective that has been decided in the initial phase of the requirement analysis is achieved successfully. After the implementation, the system provides reliable results. The system is totally menu and user friendly, which makes it easy for the users even with limited knowledge of computer environment to operate the developed system. The system avoids the drawbacks of the existing manual system and the validation facility of the system eliminates the chances of wrong data entry.

It has following features:

- User friendly
- Time saving
- Runs on any devices
- Analyzes any WhatsApp imported file.
- Accuracy
- Reliability
- Easy to use

9.2. Future work

For future work, one potential direction to enhance the WhatsApp Chat Analyzer is the implementation of sentiment analysis. Sentiment analysis aims to extract emotions and opinions expressed in chat conversations, providing valuable insights into the overall sentiment of the participants. By integrating sentiment analysis algorithms into the existing framework, the analyzer can categorize messages as positive, negative, or neutral, allowing for a deeper understanding of the emotional dynamics within the chats. This addition could open new avenues for research, such as studying the correlation between sentiment patterns and user behavior or analyzing how sentiment evolves over time within a chat group. Furthermore, incorporating sentiment analysis could enable the identification of emotional triggers or events that impact the overall tone of the conversation. By expanding the WhatsApp Chat Analyzer to include sentiment analysis, researchers can gain a more comprehensive understanding of the emotional aspects of user interactions and uncover hidden patterns that contribute to the overall dynamics of WhatsApp chats.

REFERENCES

[1] WhatsApp, "About WhatsApp," 20 April 2019. [Online]. Available: https://www.whatsapp.com/about/.

[2] W. Bani, "Whatsapp," 22 April 2019. [Online]. Available: https://en.welibani.org/weli/WhatsApp

[3] Shaikh Mohd Saqib, Prof. Sujata Bhosle (May 2022) “WHATSAPP CHAT ANALYZER”. Available:https://www.irjmets.com/uploadedfiles/paper//issue_5_may_2022/24119/final/fin_irjmets1653279784.pdf .

[4] PyCharm, "PyCharm Community," 15 December 2022. [Online]. Available: https://www.jetbrains.com/pycharm/guide/tips/compare-with-branch/ .

[5] Anaconda “Anaconda Learning” 15 May 2022 [Online], Available: https://learning.anaconda.cloud/ .

[6] Ravishankara K, Dhanush, Vaisakh, Srajan I S (May 2020) “Whatsapp Chat Analyzer”. Available: https://www.researchgate.net/publication/341876645_Whatsapp_Chat_Analyzer

PERSONAL DETAILS

Name: Dinkar Pandey
Enrollment No.: 191B099
Branch: Computer Science and Engineering
Email ID: dinkarpandey1000@gmail.com
Address: A-16 Residential complex
university of jammu J&K **Pin:** 180006
University: Jaypee University of
Engineering and Technology, Guna
Contact: +91-9906817149

Name: Harsh Marolia
Enrollment No.: 191B115
Branch: Computer Science and Engineering
Email ID: maroliaharsh@gmail.com
Address: 73/21 Collector Ganj Kanpur, Uttar
Pradesh, **Pin:** 208001
University: Jaypee University of
Engineering and Technology, Guna
Contact: +91-7234057571

Name: Yash Roopchandani
Enrollment No.: 191B293
Branch: Computer Science and Engineering
Email ID: yash21211@gmail.com
Address: D212 Shrinathpuram D, Kota,
Rajasthan **Pin:** 324010
University: Jaypee University of
Engineering and Technology, Guna
Contact: +91-7976604385

Name: Nikhil Mishra
Enrollment No.: 191B314
Branch: Computer Science and Engineering
Email ID: nikhilm0311@gmail.com
Address: B 503 Shrinath Orchid Malaroad,
Kota **Pin:** 324002
University: Jaypee University of
Engineering and Technology, Guna
Contact: +91-7878489267