TWITTER SENTIMENT ANALYSIS IN DIFFERENT LOCATIONS

A DESIGN PROJECT REPORT

Submitted by

DENNIS CYRUS J (811722104027)

HARRISH RAGHAVENDAAR RR(811722104051)

JEEVAN T (811722104063)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

November, 2024

K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report titled **"TWITTER SENTIMENT ANALYSIS IN DIFFERENT LOCATIONS"** is the Bonafide work of the students DENNIS CYRUS J (811722104027), HARRISH RAGHAVENDAAR RR (811722104051), JEEVAN T(811722104063) who carried out the project under my supervision. Certified further, that to the best of my knowledge the workreported herein does not form part of any other project report or dissertation on thebasis of which a degree or award was conferred on an earlier occasion on this orany other candidate.

SIGNATURE	**SIGNATURE**
Dr. A. Delphin Carolina Rani. M.E, Ph.D**.,**	Mrs. K. Valli Priya Dharshini. M.E., (Ph.D.,)
HEAD OF THE DEPARTMENT	**SUPERVISOR**
PROFESSOR	**ASSISTANT PROFESSOR**
Department of CSE	Department of CSE
K. Ramakrishnan College of	K. Ramakrishnan College of
Technology(Autonomous)	Technology(Autonomous)
Samayapuram – 621 112	Samayapuram – 621 112

Submitted for the viva-voce examination held on …………….

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

We jointly declare that the project report on **"TWITTER SENTIMENT ANALYSIS IN DIFFERENT LOCATIONS"** is the result of original workdone by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the award of Degree of **BACHELOR OF ENGINEERING**.

Signature

DENNIS CYRUS J

HARRISH RAGHAVENDAAR R R

JEEVAN T

Place: Samayapuram

Date:

ACKNOWLEDGEMENT

It is with great pride that we express our gratitude and in-debt to our institution "**K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)**", for providing us with the opportunity to do this project.

We are glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

We would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

We would like to thank **Dr. N. VASUDEVAN, M.Tech, Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

We heartily thanks to **Dr. DELPHIN CAROLINA RANI, M.E, Ph.D.,** Head of the department, **COMPUTER SCIENCE AND ENGINEERING** for providing her encourage pursuing this project.

I express my deep and sincere gratitude to our beloved project guide **Mrs.VALLI PRIYADHARSHINI,M.E,(Ph.D.,)** Department of **COMPUTER SCIENCE AND ENGINEERING**, for her incalculable suggestions, creativity, assistance and patience which motivated me to carry out this project.

I render my sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express my special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

ABSTRACT

Social media platforms like Twitter have become a crucial data source for understanding public attitudes and emotional trends on mental health issues. This project leverages Twitter's extensive data streams to analyze sentiments around mental health discussions across different countries.

By integrating advanced **Natural Language Processing (NLP)** techniques and sentiment analysis, tweets are classified into categories such as "**happy**," "**depressed**," and "**neutral**," offering insights into the emotional landscape of mental health discourse.

The backend of the system is built using Flask, enabling seamless integration with the Twitter API to fetch real-time tweets based on user-defined country inputs. A SQLite database stores the processed tweets to reduce redundant API calls and ensure faster response times.

Sentiment analysis is performed using the **TextBlob** library, which assesses the polarity of tweet content to classify sentiments. The processed data is visualized through dynamically generated bar charts, giving users an intuitive understanding of sentiment distributions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION	FULL FORM
NLP	Natural Language Processing
JS	Java Script
UI/UX	User Interface/User Experience
SQL	Structured Query Language
API	Application Programming Interface
ORM	Object-Relational Mapping
DFD	Data Flow Diagram
UML	Unified Modeling Language
DBMS	Database Management System
ML	Machine Learning
JSON	JavaScript Object Notation
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
JWT	JSON Web Token

CHAPTER 1

INTRODUCTION

Social media platforms, such as Twitter, have significantly impacted the way individuals express opinions, share personal experiences, and engage in public discussions, particularly around sensitive topics like mental health. Mental health, an essential element of overall well-being, is often subject to stigma and misunderstandings. The open and accessible nature of platforms like Twitter offers a unique opportunity to examine how mental health is portrayed, how people describe their emotional states, and the broader societal attitudes toward these issues.

Utilizing sentiment analysis, a natural language processing (NLP) technique, this project aims to analyze tweets related to mental health on a global scale. Sentiment analysis categorizes text into emotional tones, such as positive, negative, or neutral, enabling the extraction of valuable insights from vast amounts of social media data. By applying this method to Twitter content, the study seeks to uncover patterns and trends in public sentiment regarding mental health across various countries and demographics, shedding light on the perceptions and discourse surrounding the topic.

The methodology integrates multiple technologies, including Flask for backend services, Tweepy for accessing the Twitter API, and TextBlob for sentiment analysis. By collecting and analyzing tweets, the project classifies them based on their sentiment, with results visualized through accessible bar charts, providing an intuitive understanding of public sentiment on mental health.

1.1 PROJECT OVERVIEW

The **Mental Health Sentiment Analysis Using Twitter Data** project is designed to analyze public discussions, attitudes, and emotional trends related to mental health across different countries using Twitter as a primary data source

1.1.1 Backend Overview

1. **Data Retrieval**:
   - The backend fetches tweets using the Twitter API via the Tweepy library. A query is structured to retrieve recent tweets related to mental health in specific countries while excluding retweets for better accuracy.

2. **Data Storage and Management**:
   - An SQLite database is used to store tweets and their corresponding sentiment classifications. This ensures data persistence and reduces redundant API calls.

3. **Sentiment Analysis**:
   - The **TextBlob** library evaluates the polarity of tweet text. Tweets are classified into three categories:
     - **Happy**: Positive sentiment.
     - **Depressed**: Negative sentiment.
     - **Neutral**: Neutral or balanced sentiment.

1.1.2 Frontend Overview

1. **Interactive User Interface**:
   - A simple and intuitive web interface enables users to select a country and trigger the sentiment analysis process.

2. **Dynamic Results Display**:
   - The results are presented as bar charts directly on the webpage, showcasing the sentiment distribution of tweets for the selected country.

3. **Error Handling**:
   - The interface handles scenarios such as no tweets found, API rate limits, or invalid inputs, ensuring smooth user interaction.

1.2 PROBLEM STATEMENT

Mental health is a critical global issue, yet it remains surrounded by stigma and misinformation in many regions. Understanding public sentiment and attitudes towards mental health is essential for fostering awareness, reducing stigma, and implementing effective support systems. Social media platforms like Twitter offer a vast repository of real-time, user-generated data that reflects public opinions and emotional states.

1. **Identifying Relevant Data**: Extracting meaningful discussions about mental health from the overwhelming volume of tweets.

2. **Classifying Sentiments**: Determining whether public sentiment is positive, negative, or neutral to uncover emotional trends.

1.2.1 GOALS

The goals of the EAAP project are as follows:

- By analyzing tweets, the project seeks to classify public opinions into positive, negative, or neutral sentiments.

- The project focuses on regional trends by analyzing tweets from different countries

- Leveraging Twitter's live data stream, the project allows for real-time tracking of mental health discussions.

- The system is designed to be extensible, allowing it to be adapted for analyzing sentiments in other fields like politics, healthcare, or environmental issues

1.3 OBJECTIVE OF THE PROJECT

- Utilize sentiment analysis techniques to classify tweets related to mental health into categories such as positive, negative, or neutral.

- Offer country-specific sentiment analysis by analyzing tweets from various regions, helping to understand how mental health is discussed differently across different cultural and geographical contexts.

- Enable real-time monitoring of mental health discussions on Twitter, allowing stakeholders to track shifts in public sentiment in response to global events.

1.4 SCOPE OF THE PROJECT

1.4.1 Data Collection

- **Source**: The project focuses exclusively on Twitter as a data source. Tweets related to mental health topics will be retrieved using the Twitter API, specifically targeting public discussions that include keywords related to mental health and excluding retweets

1.4.2 Sentiment Analysis

- **Methodology**: The sentiment of the collected tweets will be analyzed using natural language processing (NLP) techniques. TextBlob Tweets will be classified as "happy," "depressed," or "neutral" based on their sentiment polarity.

1.4.3 Data Storage

- **Database**: The project will use an **SQLite** database to store the fetched tweets and their corresponding sentiment classifications. This ensures persistence and avoids re-fetching the same data repeatedly.

1.4.4 Visualization

- **Chart Generation**: The project will generate visual representations of the sentiment distribution using Matplotlib. These will be presented as bar charts showing the number of tweets categorized under each sentiment .

CHAPTER 2

LITERATURE SURVEY

2.1 TITLE: A DOMAIN-AGNOSTIC NEUROSYMBOLIC APPROACH FOR BIG SOCIAL DATA ANALYSIS:EVALUATING MENTAL HEALTH SENTITEMENT ON SOCIAL MEDIA DURING COVID-19

AUTHOR:KHANDELWAL,V.,GAUR,M.,KURSUNCU,U.,SHALIN,V.,&SHETH,A.
YEAR: 2024

This study presents a neurosymbolic approach for analyzing sentiment related to mental health issues, particularly depression and anxiety, on social media platforms during the COVID-19 pandemic. The model proposed by the authors integrates neural networks with symbolic knowledge to improve the interpretation and detection of mental health-related sentiments in tweets. This methodology addresses the challenge of interpreting rapidly evolving language and slang, which is common in social media interactions.

The hybrid model combines machine learning techniques with symbolic reasoning, allowing it to better understand the context and sentiment behind social media content, particularly when slang or new terms emerge.By focusing on COVID-19-related tweets, the research highlights the need for dynamic tools that can quickly adapt to changes in language and social context during unprecedented events like a pandemic.

2.2 TITLE: ENHANCING MENTAL HEALTH AWARNESS THROUGH TWITTER ANALYSIS: A COMPARTIVE STUDY OF MACHINE LEARNING AND HYBRID DEEP LEARNING TECHINIQUES

AUTHOR: CHATTERJEE, R., GUPTA, R. K., GUPTA, B.

YEAR: 2023

Text sentiment analysis is mostly used for the assessment of the author's mood depending on the context. The purpose of sentiment analysis (SA) is to discover the exactness of the underlying emotion in a given situation. It has been applied to various fields, including stock market pre dictions, social media data on product evaluations, psychology, the judiciary, forecasting, illness prediction, agriculture, and more. Many researchers have worked on these topics and generated important insights. These outcomes are useful in the field because they (outcomes) help people comprehend the general summary quickly, Additionally, SA aids in limiting the harmful effects of some posts on various social media sites such as Facebook and Twitter. For these reasons and more, we are proposing an approach to filter the social media content that could be emotionally harm ful to the user, through getting the SM content(we will refer to social media as SM), for that we have used Twitter API to get the user posts (Twitter as an example of SM), then we have used API natural understanding language API tool to extract and classify the emotions of the Twitter content into five basic emotional categories: Joy, sadness, anger, fear, disgust. into an array of emotions, after that, we have defined a perfect emotion array from over 450 words from the English language. The main purpose of this comprehensive research article is to examine the proposed solution that we have conducted to improve the quality of content displayed to users emotionally

2.3 TITLE: SENTIMENT ANALYSIS ON MENTAL HEALTH CAMPAIGN TWETTS from 2017 to 2023"

AUTHOR: Z.Hama,M.O.Khan,M.Iqbal,andM.A.Anwar

YEAR: 2023

This paper investigates the sentiments expressed in tweets related to four major mental health campaigns: Mental Health Awareness Week (MHAW), Eating Disorders Awareness Week (EDAW), and others, spanning from 2017 to 2023. The authors utilized VADER (Valence Aware Dictionary and sEntiment Reasoner), a sentiment analysis tool specifically tailored for social media text, to evaluate the tone of these tweets. The study analyzes sentiment scores from different campaigns over the years, looking for trends in public perception analysis of the data revealed that the campaigns were successful in engaging a broad audience, with a notable shift in tone from neutral to positive across multiple years.The study found that over time, the general sentiment towards mental health issues became more positive, particularly during campaign periods. Tweets during Mental Health Awareness Week showed an overwhelming positive tone, reflecting growing public empathy and awareness. A emphasizes the power of social media in gauging public sentiment and how sentiment analysis tools like VADER can help track and improve mental health awareness campaigns. The research also highlights how sustained campaigns and their timely execution can positively influence public perception of mental health issues.

2.4 TITLE: SENTIMENTS ABOUT MENTAL HEALTH ON TWITTER—BEFORE AND DURING THE COVID-19 PANDEMIC

AUTHOR: FELIX BEIERL, RUDIGE PRYSS, AKIKO AIZAWA

YEAR: 2023

During the COVID-19 pandemic, the novel coronavirus had an impact not only on public health but also on the mental health of the population. Public sentiment on mental health and depression is often captured only in small, survey-based studies, while work based on Twitter data often only looks at the period during the pandemic and does not make comparisons with the pre pandemic situation. We collected tweets that included the hashtags #MentalHealth and #Depression from before and during the pandemic (8.5 months each). We used LDA (Latent Dirichlet Allocation) for topic modeling and LIWC, VADER, and NRC for sentiment analysis. We used three machine learning classifiers to seek evidence regarding an automatically detectable change in tweets before vs. during the pandemic: (1) based on TF-IDF values, (2) based on the values from the sentiment libraries, (3) based on tweet content (deep-learning BERT classifier). Topic modeling revealed that Twitter users who explicitly used the hashtags #Depression and especially #MentalHealth did so to raise awareness. We observed an overall positive sentiment, and in tough times such as during the COVID 19 pandemic, tweets with #MentalHealth were often associated with gratitude. Among the three classification approaches, the BERT classifier showed the best performance, with an accuracy of 81% for #MentalHealth and 79% for #Depression. Although the data may have come from users familiar with mental health, these findings can help gauge public sentiment on the topic.

CHAPTER 3

EXISTING SYSTEM

1. Data Collection from Twitter

- **Tweet Extraction**: Use Twitter’s API to extract tweets based on keywords related to mental health (e.g., "anxiety," "depression," "stress") and location-specific hashtags (e.g., #NYC, #London).
- **Location Data**: Incorporate geotagging to understand the geographical location of each tweet. If geotagging is unavailable, use inferred location data from users' profiles (e.g., user-reported location or time zone).
- **Time Period**: Define the time range for analysis (e.g., weekly, monthly).

2. Data Preprocessing

- **Text Cleaning**: Remove irrelevant content such as advertisements, spam, URLs, and stop words from the tweets.
- **Natural Language Processing (NLP)**: Apply NLP techniques for sentiment analysis (positive, neutral, negative).

3. Sentiment and Emotion Analysis

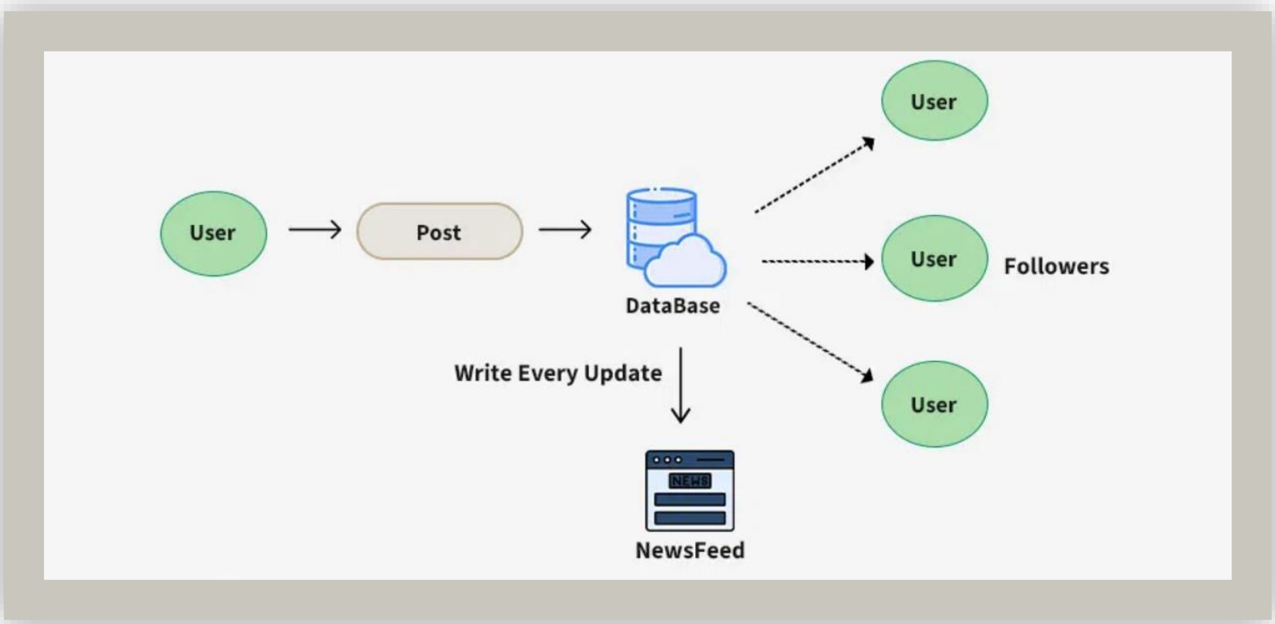
- **Sentiment Analysis**: Classify tweets into positive, neutral, or negative sentiments. This helps to gauge overall emotional tone related to mental health in various locations.
- **Emotion Recognition**: Use more detailed emotional detection, for example, detecting anger or sadness linked to mental health issues.

4. Geospatial Analysis

- **Location-based Sentiment Mapping**: Map sentiments and emotions by region or city. This can highlight regions with higher levels of negative sentiment or distress.
- **Heatmap Visualization**: Show the intensity of mental health-related discussions across different locations using a heatmap.
- **Comparing Locations**: Compare mental health trends between different geographic regions to identify patterns (e.g., urban vs. rural mental health differences).

5. Temporal Analysis

- **Trends Over Time**: Track mental health sentiment over days, weeks, or months to observe if specific events (e.g., holidays, pandemics) influence mental health discourse.
- **Seasonal Trends**: Analyze the influence of seasons or specific periods like winter months, school exams, or holidays on mental health discussions.



CHAPTER 4

PROPOSED SYSTEM

The proposed system is designed to monitor and analyze mental health trends by analyzing Twitter data across different locations. It will collect real-time tweets containing mental health-related keywords and geolocation data using the Twitter API. The system will preprocess the text data, applying natural language processing (NLP) techniques to clean and analyze the content for sentiment and emotion. Sentiment analysis will classify tweets into positive, negative, or neutral categories, while emotion detection will identify feelings such as sadness, anger, or anxiety.

Additionally, the system will use geospatial analysis to map these sentiments to specific locations and create heatmaps to visualize mental health trends geographically. Temporal analysis will track mental health sentiment over time, identifying patterns or seasonal variations.

Furthermore, the system will provide personalized mental health resource recommendations based on the trends identified, supporting targeted interventions in specific regions. This system aims to be an effective tool for proactive mental health monitoring, offering valuable data for policymakers.

Key Features of the Proposed System:

A. Real-Time Monitoring:

- The system will provide up-to-the-minute updates on mental health sentiment across locations, allowing for timely interventions.

B. Geospatial & Temporal Insights:

- Geospatial heatmaps and temporal trend analysis will provide a comprehensive view of where and when mental health concerns are peaking.

C. Emotional and Sentimental Insights:

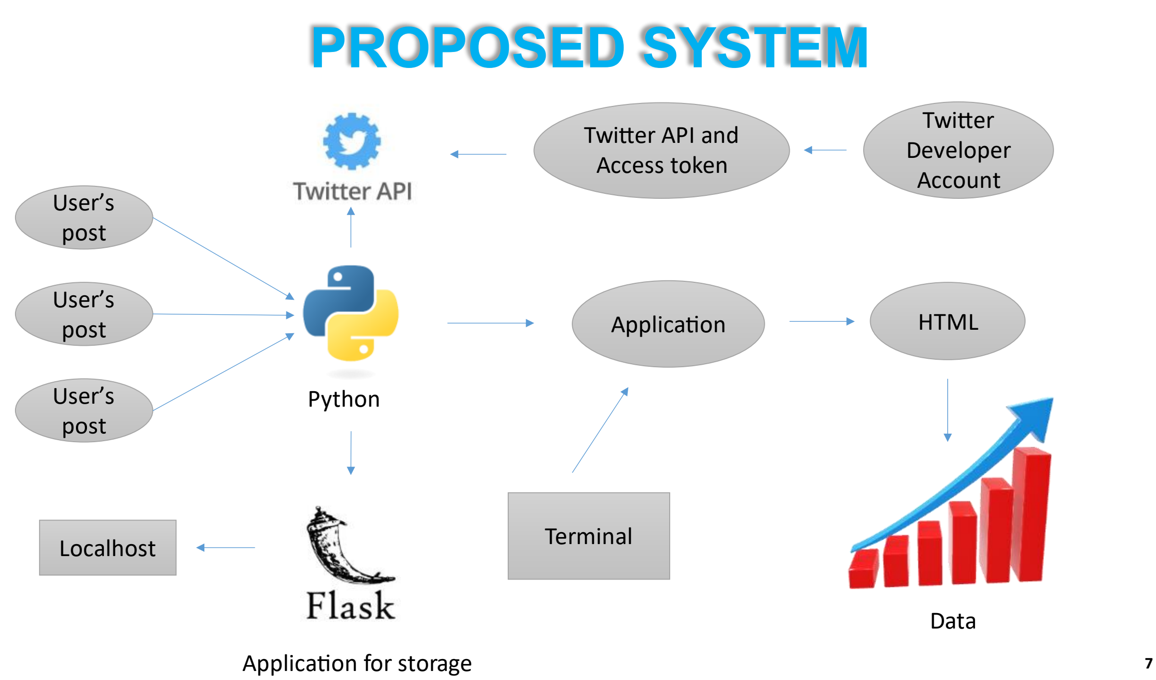
- The system will go beyond basic sentiment analysis, offering emotional insights into how people are feeling in different areas and during different times.

D. Localized Support:

- The system will highlight areas with the highest mental health distress and suggest resources tailored to specific geographic locations or concerns.

E. Proactive Public Health Monitoring:

- By detecting mental health issues as they arise, the system can alert health organizations and local governments to take action before crises escalate.



CHAPTER 5

SYSTEM ARCHITECTURE

Sentiment analysis is performed to categorize tweets as positive, negative, or neutral, while emotion detection models identify deeper emotional tones such as sadness, anger, and frustration. The system also incorporates geospatial analysis to map these sentiments to specific regions, creating heatmaps that visually highlight areas with the highest levels of mental health distress. Temporal analysis is used to track changes in mental health sentiment over time, helping to identify patterns or correlations with events such as holidays, social issues, or public crises.

The system generates real-time alerts and actionable insights based on these findings, allowing local authorities, mental health organizations, and policymakers to intervene when necessary. Additionally, it can provide recommendations for mental health resources in regions with high negativesentiment.

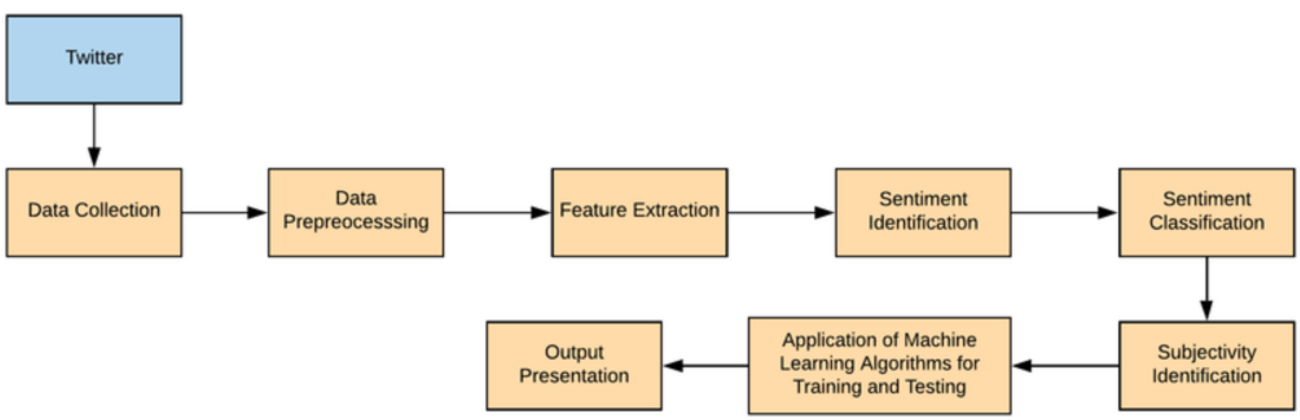


Fig 5.1 System Design

5.1 DATA FLOW DIAGRAM

Mental Health Monitoring and Analysis system outlines the flow of data through various stages of the system, from data collection to processing, analysis, and reporting. Initially, the system collects real-time tweets using the Twitter API, which is filtered based on mental health-related keywords and location-specific hashtags. These tweets, along with metadata like geolocation and user information, are sent to the Data Collection Layer.

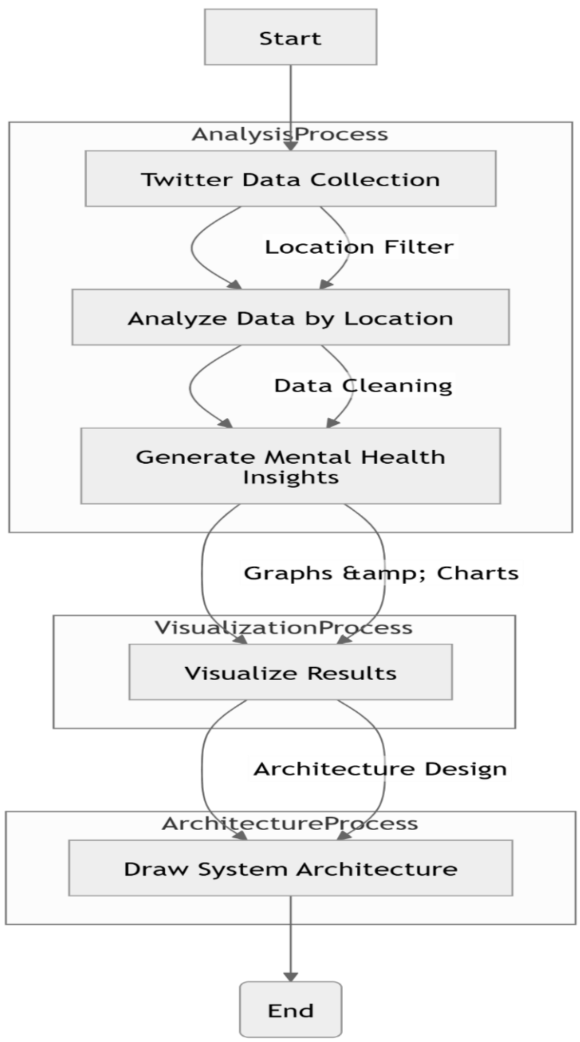


Fig 5.2 Data Flow Diagram

USER LOGIN:

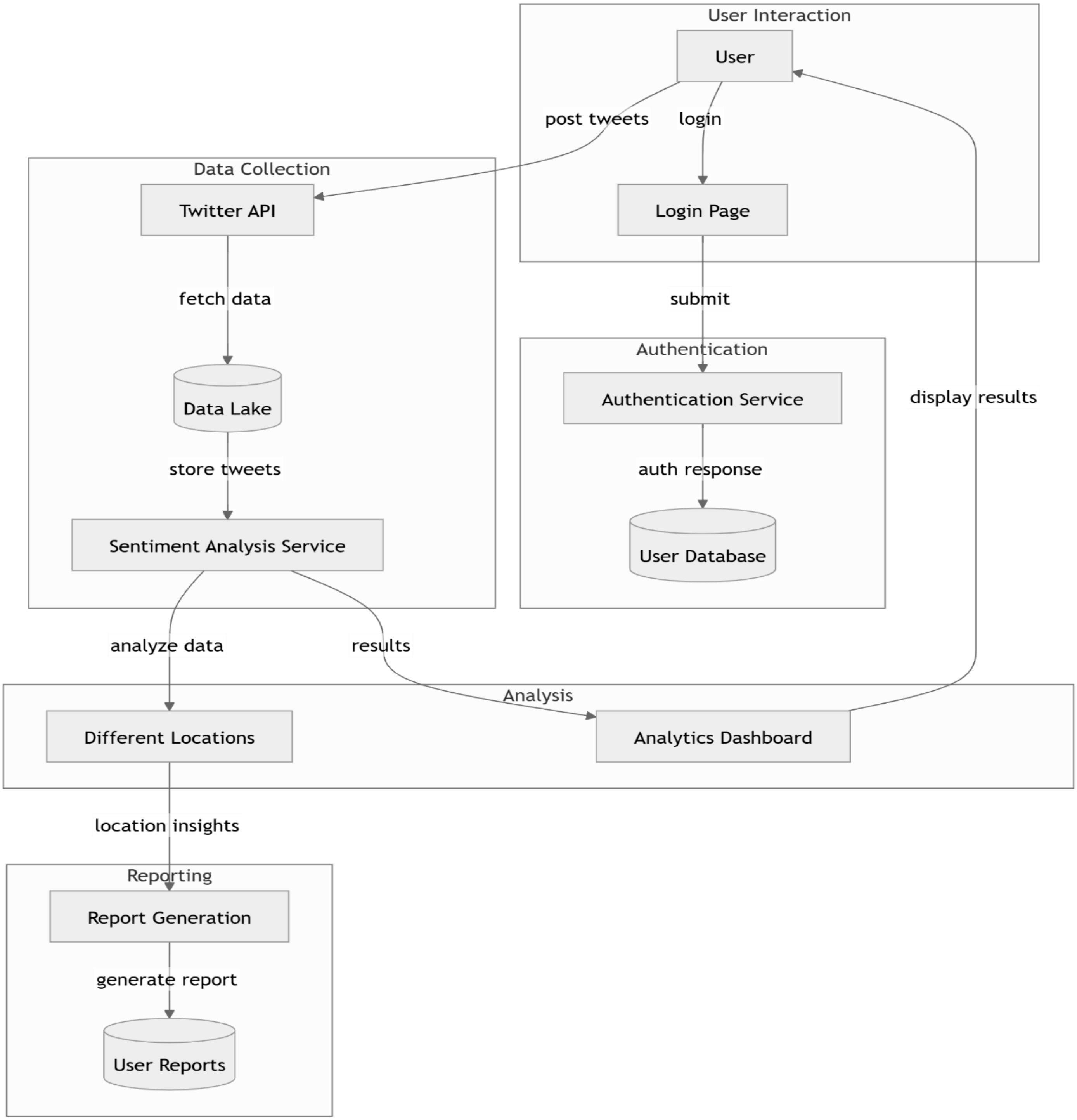


Fig 5.3 Login Process

5.2 USE CASE DIAGRAM

Mental Health Monitoring and Analysis system outlines the interactions between users (actors) and the system's functionalities (use cases). In this diagram, the primary actors include **Mental Health Organizations**, **Government Agencies**, **Local Authorities**, and **System Administrators**, each having specific roles and interactions with the system.

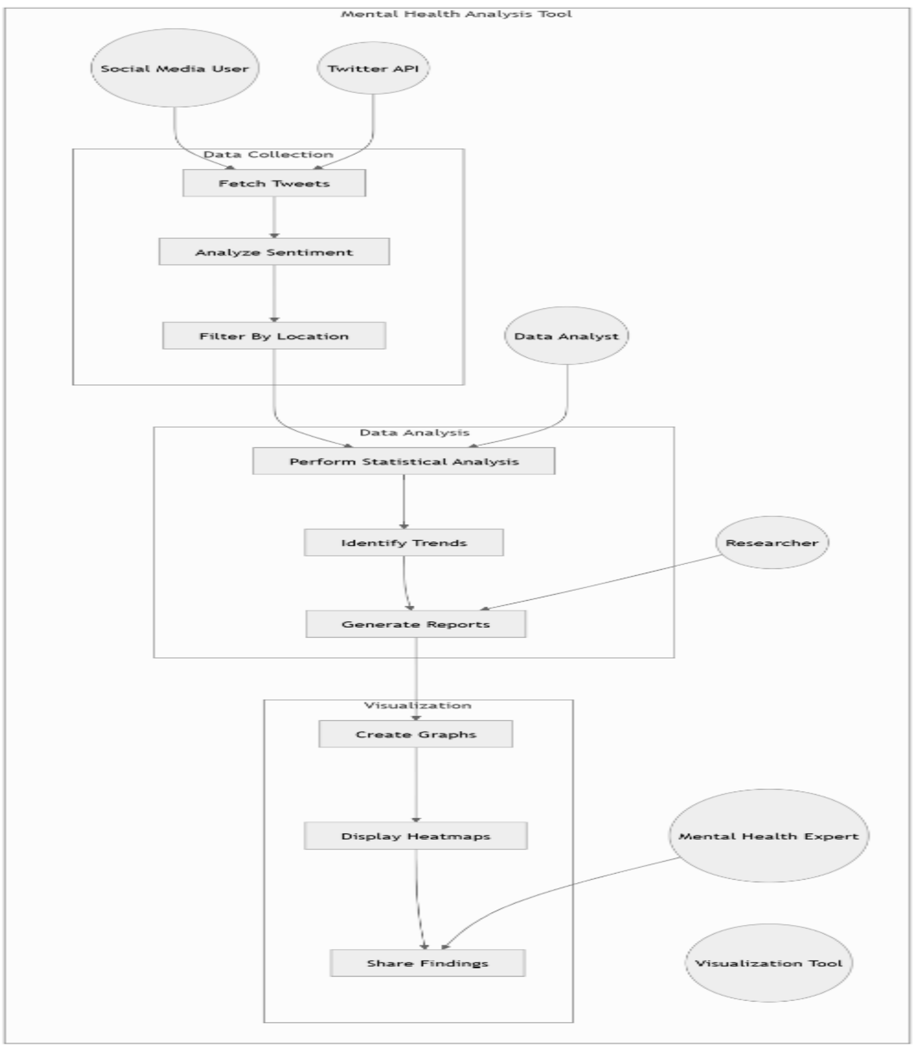


Fig 5.4 Use Case Diagram

5.3 ACTIVITY DIAGRAM

Mental Health Monitoring and Analysis system represents the flow of activities or tasks that the system performs to process and analyze Twitter data related to mental health. It also illustrates how the system interacts with various stakeholders (actors) to provide actionable insights.

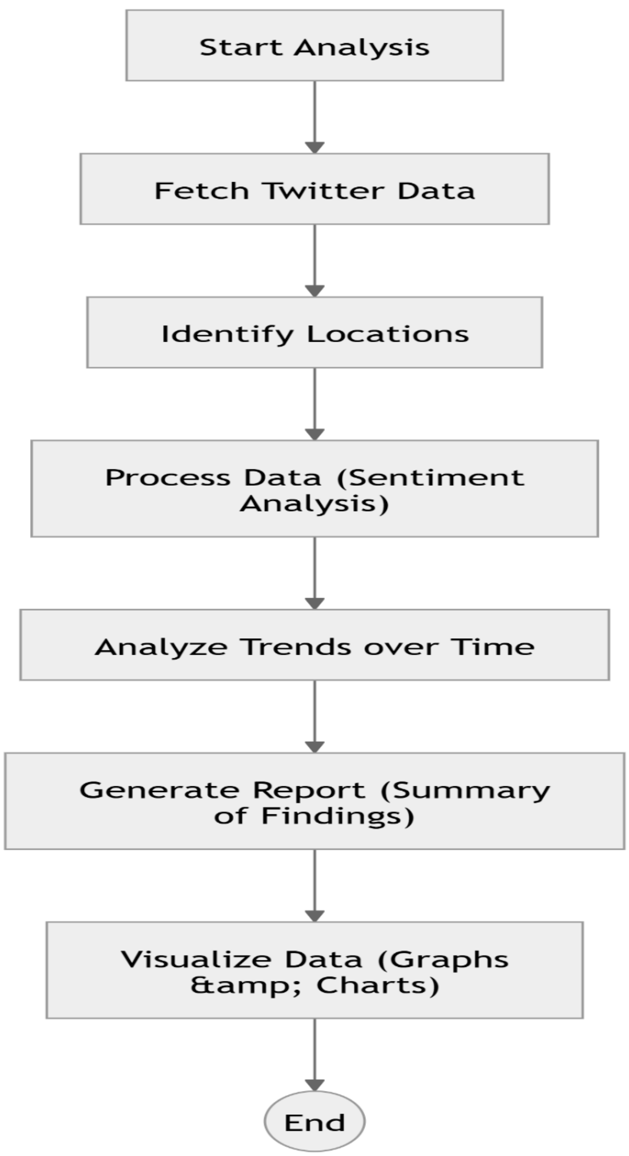


Fig 5.5 Activity Diagram

5.4 SEQUENCE DIAGRAM

Mental Health Monitoring and Analysis system outlines the interaction between the primary components of the system: Mental Health Organizations, the System, and the Twitter API. The process begins when a Mental Health Organization requests data from the system. The system then connects to the **Twitter API to collect real-time tweets, based on keywords related to mental health (such as "depression" or "anxiety") and location-specific hashtags. The collected tweets, along with relevant metadata (e.g., geolocation, timestamps), are returned to the system.

Once the data is collected, the system preprocesses the raw tweets by cleaning them (removing irrelevant content) and performing text normalization (tokenization, lemmatization). After preprocessing, the system conducts sentiment and emotion detection to classify the tweets as positive, negative, or neutral, and also identifies underlying emotions like sadness or anger. The results of this sentiment and emotion analysis are then made available to the Mental Health Organization.

Furthermore, the system can generate and store detailed reports, providing insights into regional mental health trends and creating alerts in response to identified spikes in negative sentiment or distress. These insights and alerts are accessible to Mental Health Organizations, who can use the data to make informed decisions about resource allocation and intervention strategies. The entire process ensures continuous, real-time monitoring and analysis of mental health trends based on social media data.

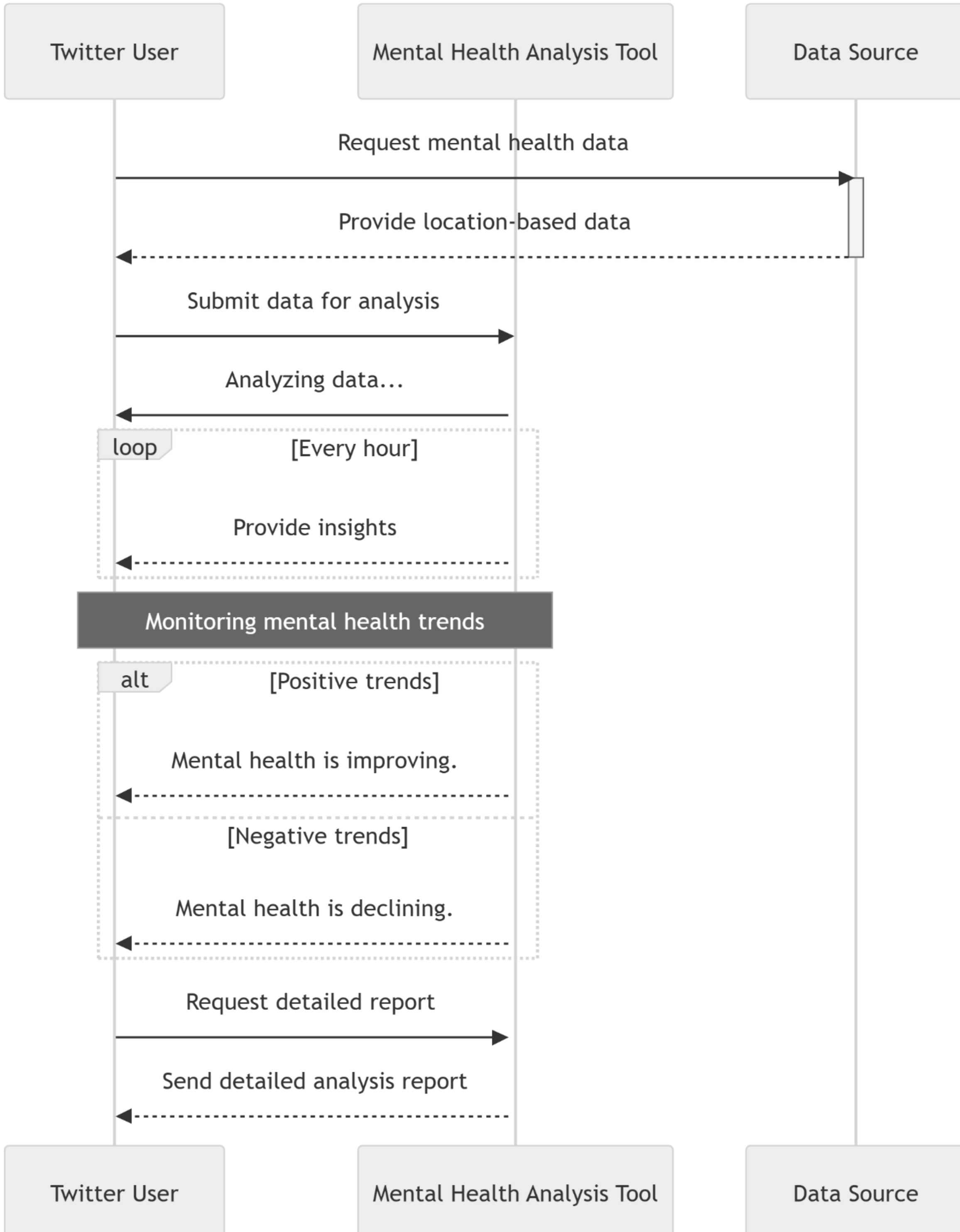


Fig 5.6 Sequence Diagram

5.5 DATABASE DESIGN:

Mental Health Monitoring and Analysis System is structured to efficiently store and manage large volumes of data generated by real-time tweet collection, sentiment analysis, emotion detection, and geospatial and temporal tracking. The system uses **PostgreSQL** as the database management system (DBMS) for its scalability, reliability, and support for complex queries. The **Prisma ORM** is used for simplified database interaction between the backend and the database, ensuring seamless data operations and flexibility.

The schema design is centered around key entities such as **users**, **tweets**, **sentiment analysis results**, **emotion data**, **geospatial information**, and **alerts**, with appropriate relationships established to enable quick retrieval and accurate reporting. Data normalization principles are applied to eliminate redundancy and maintain consistency, ensuring that storage is optimized and query performance is fast and efficient.

The database design goals are:

- **To securely store interrelated data** while minimizing inconsistencies and ensuring data integrity.

- **To enable quick and flexible access** to data, providing the ability to scale with large volumes of tweet data and analytical results.

- **To establish meaningful relationships between entities**, allowing for detailed and comprehensive queries, such as identifying trends in specific regions, analyzing temporal changes in sentiment, and generating alerts based on negative sentiment or emotional distress.

5.5.1 USER ACCOUNT CREATION

The **User** table handles the creation and management of user accounts in the Mental Health Monitoring and Analysis System. It stores essential details like email, username, password hash, role, and timestamps for account creation. This table is critical for ensuring secure user authentication, and managing the roles of various stakeholders, such as mental health organizations, local authorities, or administrators.

Field	**Description**	**Constraints**
id	Unique identifier for each user	Primary Key, Auto Increment
email	User's email address	Unique, Not Null
username	User's chosen username	Unique, Not Null
passwordHash	Encrypted password for secure login	Not Null
role	Role of the user (e.g., Admin, Mental Health Org, Local Authority)	Not Null, Default: 'User'
createdAt	Timestamp for account creation	Default: now()
updatedAt	Timestamp for when the account was last updated	Default: now(), on update: now()

Table 5.1 User account creation

5.5.2 ADMINISTRATOR LOGIN

The **User** table also handles **administrator accounts** with elevated privileges. While the schema does not explicitly differentiate between regular users and administrators, roles and permissions can be implemented programmatically within the system to assign special administrative capabilities to users with specific roles. The role of an administrator can be determined by the role field in the table, and administrators are granted access to advanced system management features.

Field	**Description**	**Constraints**
id	Unique identifier for each administrator	Primary Key, Auto Increment
email	Administrator's email address	Unique, Not Null
username	Administrator's username	Unique, Not Null
passwordHash	Encrypted password for secure login	Not Null
role	Role of the user, which determines if they are an Admin	Not Null, Default: 'User'
createdAt	Timestamp for account creation	Default: now()
updatedAt	Timestamp for when the account was last updated	Default: now(), on update: now()

TABLE 5.2 Administrator Login

CHAPTER 6

SYSTEM REQUIREMENTS

6.1 SOFTWARE REQUIREMENTS

- Operating System: Windows 10 or Higher
- Coding Language: HTML, CSS, TypeScript, Prisma
- Tool: Visual Studio Code
- Database: PostgreSQL
- Flask**:**It will handle user requests, API calls, and serve the front-end content.

6.2 HARDWARE REQUIREMENTS**

- Processor: Intel Core i3 or Higher
- Hard Disk: Not Applicable (Web Application)
- Monitor: Responsive to all Screen Sizes (Responsive Design)
- RAM: Minimum 2GB

6.3 HARDWARE DESCRIPTION

Windows 10 is a modern operating system developed by Microsoft, designed to run efficiently on a broad range of hardware configurations. Known for its performance, reliability, and security features, Windows 10 is well-suited for both development and testing environments.

6.4 SOFTWARE DESCRIPTION:

6.4.1 Visual Studio Code

Visual Studio Code (VS Code) is a lightweight yet powerful source code editor widely used for web and backend development. It offers robust support for modern programming languages like TypeScript,Node.js and JavaScript along with a comprehensive ecosystem of extensions. Developers can efficiently write, debug, and deploy code using features like IntelliSense, Git integration, and an integrated terminal. For the Mental Health Monitoring and Analysis System, VS Code is utilized to develop both the front-end and back-end, managing the platform's modular architecture with seamless integration of tools.

6.4.2 TypeScript

TypeScript is a strongly-typed superset of JavaScript that helps developers catch errors during development, enhancing code quality and maintainability. By adding static types, TypeScript ensures applications are robust and scalable. For the Mental Health Monitoring and Analysis System, TypeScript is used to build modular components and maintain a scalable architecture for handling dynamic features such as sentiment analysis and user interaction.

6.4.3 Next.js

Next.js is a React-based framework for building server-side rendering (SSR) and static web applications. It improves performance with optimized page loading, automatic code-splitting, and built-in API routes. In the Mental Health Monitoring and Analysis System, Next.js is used to render the front-end interface dynamically, enabling smooth interaction with live sentiment trends, geospatial visualizations, and real-time mental health analytics.

6.4.4 PostgreSQL

PostgreSQL is an advanced open-source relational database management system renowned for its reliability and scalability. The Mental Health Monitoring and Analysis System uses PostgreSQL to store and manage user accounts, tweets, sentiment analysis results, and alert data securely. Its seamless integration with Prisma ORM ensures efficient queries, enabling real-time data analysis and secure storage.

6.4.5 Prisma

Prisma is a modern Object-Relational Mapping (ORM) tool that simplifies database access by providing type-safe queries. Its integration with TypeScript and PostgreSQL makes it ideal for managing the back-end of the Mental Health Monitoring and Analysis System. Prisma reduces the risk of inconsistencies while enabling rapid development of scalable, maintainable database schemas.

6.4.6 Node.js and NPM Modules

Node.js serves as the runtime environment for the back-end of the system, offering a non-blocking, event-driven architecture that efficiently handles multiple concurrent requests. With its package manager (NPM), Node.js provides access to thousands of modules, enhancing the platform's functionality.

6.5 SCALABILITY AND SECURITY

6.5.1 Scalability

The Mental Health Monitoring and Analysis System is designed with a modular and extensible architecture to support future growth and enhancements. By leveraging tools like PostgreSQL and Prisma, the platform can efficiently handle increasing data volumes and user activity without performance degradation. Features such as dynamic API routes and server-side rendering (via Next.js) allow seamless scaling of both front-end and back-end components.

6.5.2 Security

Security is a key priority for the system. User authentication and data protection are implemented using industry-standard practices, such as password hashing with bcrypt and secure token management using JWT. APIs are designed with robust encryption mechanisms to ensure safe communication between the client and server. Additionally, regular dependency updates and vulnerability checks are conducted to maintain a secure development environment.

6.6 FUTURE WORK

1. Mobile Application Development: Extend the platform's reach by developing a dedicated mobile application for better accessibility and convenience.

2. AI-Driven Insights: Implement machine learning algorithms to provide personalized mental health insights, trend forecasting, and enhanced data analytics.

3. Gamification Features: Introduce user engagement strategies such as rewards, achievements, and leaderboards to encourage active participation in mental health initiatives.

CHAPTER 7

SYSTEM TESTING

The purpose of system testing is to ensure the software functions as expected, identifying and resolving errors while verifying that both functional and non-functional requirements are met. This process evaluates the system's components and their integration to ensure reliability and performance. System testing for the **Mental Health Monitoring and Analysis System** follows modern testing methodologies to guarantee accuracy, security, scalability, and user satisfaction.

7.1. TYPES OF TESTS

7.1.1 UNIT TESTING

Unit testing validates individual components to ensure each module works correctly in isolation. It includes backend and frontend validations.

- **Backend Testing**: Functions for authentication (using bcrypt and JWT) were tested for secure password hashing and token generation.
- **Frontend Testing**: React components (via Next.js) were validated for proper rendering and state management

7.1.2 INTEGRATION TESTING

Integration testing ensured seamless communication between the backend API, PostgreSQL database (via Prisma ORM), and the Next.js frontend.

Key Scenarios:

- User workflows like registration, login, and token validation.
- Data retrieval and manipulation using Prisma.
- Interactions between frontend UI and backend APIs for dynamic actions.

7.1.3 SYSTEM TESTING

System testing verified that the integrated system met all specified requirements.

Key Tests:

- Cross-browser compatibility on Chrome, Firefox, and Edge.
- API response times and database accuracy under heavy loads.
- Security checks for password encryption and token validation.

7.1.4 WHITE BOX TESTING

White Box Testing focused on validating internal logic and code structures.

Key Areas:

- Authentication workflows with bcrypt and JWT.
- Backend data validation and error handling.
- Static type enforcement with TypeScript.

7.1.5 BLACK BOX TESTING

Black Box Testing validated the system without accessing the internal code, focusing on end-user experience.

- Ensuring smooth user registration and login flows.
- Testing frontend responsiveness and accessibility across devices.
- Checking error messages and dynamic updates in the UI.

7.1.6 OUTPUT TESTING

- Output testing verified that the system provided accurate and reliable results.
- JSON responses from APIs for correctness.
- Rendered HTML and CSS elements for visual consistency.

7.1.7 USER ACCEPTANCE TESTING (UAT)

UAT confirmed the platform met user needs in real-world scenarios.

Scenarios:

- Registration, login, and secure user sessions.
- Smooth navigation and accurate data visualization.
- User-friendly prompts and error handling.

7.1.8 AUTOMATED TESTING TOOLS

Automated tools streamlined the testing process:

- **Jest**: Unit and integration testing for TypeScript code.
- **Postman**: API endpoint testing to ensure proper backend communication.
- **ESLint and Prettier**: Code quality assurance for adherence to standards.

7.1.9 SECURITY TESTING

Security testing prioritized safeguarding sensitive user data:

- bcrypt ensured password hashes are strong against attacks.
- JSON Web Tokens (JWT) were tested for secure session management.
- Helmet secured HTTP headers to mitigate vulnerabilities like XSS.

7.1.10 PERFORMANCE TESTING

Performance testing validated the system under heavy loads and concurrent users:

- **API Performance**: Verified acceptable response times under stress.
- **Database Testing**: Ensured PostgreSQL efficiently handled large datasets.
- **Frontend Testing**: Assessed responsiveness and rendering during peak usage.

7.1.11 RESULTS

The testing process confirmed that the system:

- Meets all functional and non-functional requirements.
- Provides a secure, scalable, and reliable platform for users.
- Is user-friendly, with intuitive navigation and fast response times.

CHAPTER 8

CONCLUSION AND FUTURE WORK

Conclusion:

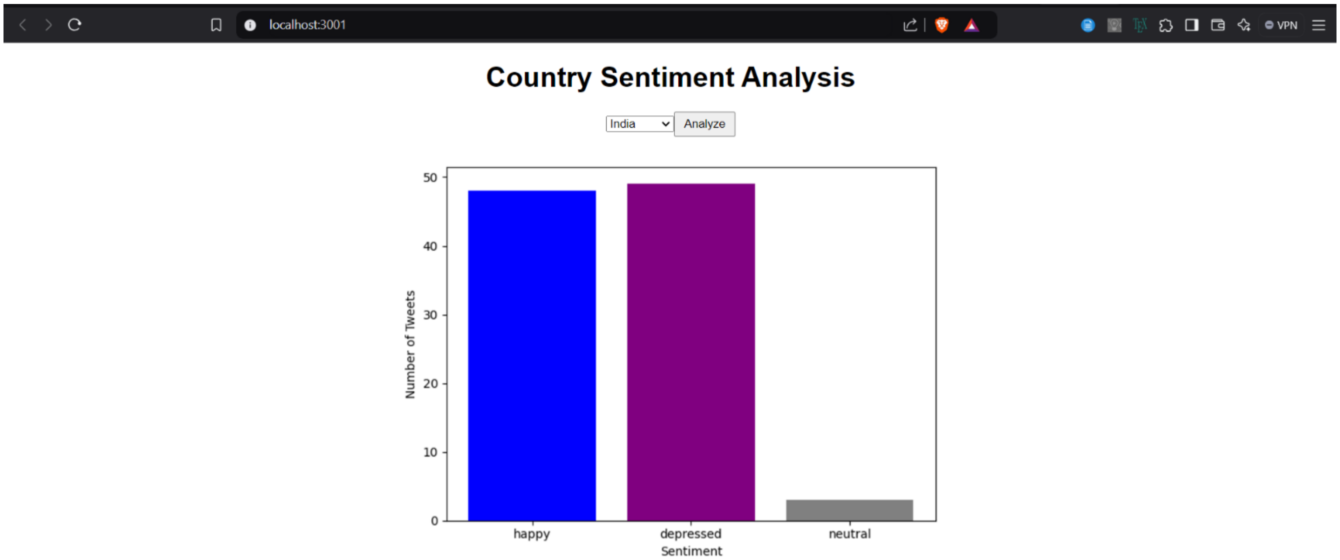
The **Mental Health Monitoring and Analysis System** is designed to assess public mental health trends, analyze sentiment in social media data, and provide actionable insights. The platform empowers users by offering real-time geospatial visualizations, sentiment analysis, and personalized mental health resources. It bridges the gap between raw data and meaningful analysis to address mental health challenges at a community level.

The system uses a robust tech stack, including Next.js, PostgreSQL, Prisma, TypeScript, and Node.js, to ensure high performance and scalability. Secure authentication via bcrypt and JWT, combined with automated testing, guarantees reliability. The platform's modular architecture and user-friendly design make it adaptable for future enhancements.

Future Work:

- Mobile Application Development: Expanding to mobile platforms for broader accessibility.
- AI-Driven Insights: Using machine learning to personalize mental health resources.
- Real-Time Data Integration: Incorporating live APIs for dynamic updates in sentiment trends.

APPENDIX A


localhost:3001
VPN
Country Sentiment Analysis
India
Analyze
50
40
30
20
10
0
Number of Tweets
happy
depressed
neutral
Sentiment

APPENDIX B

SOURCE CODE:

app.py

```
from flask import Flask, request, jsonify, send_file
from flask_cors import CORS
import tweepy
import sqlite3
from textblob import TextBlob
import matplotlib.pyplot as plt
import io
import time
from datetime import datetime, timedelta

app = Flask(_name_)
CORS(app)

# Twitter API credentials
BEARER_TOKEN = "YOUR_TWITTER_BEARER_TOKEN"
client = tweepy.Client(bearer_token=BEARER_TOKEN)

# SQLite database connection function
def get_db_connection():
    conn = sqlite3.connect('tweets.db')
    return conn

# Create database table if it doesn't exist
```

```
def init_db():
    conn = get_db_connection()
    conn.execute('''CREATE TABLE IF NOT EXISTS tweets
                    (id INTEGER PRIMARY KEY AUTOINCREMENT,
                     text TEXT,
                     sentiment TEXT,
                     country TEXT)''')
    conn.commit()
    conn.close()

# Track last request time
last_request_time = None
RATE_LIMIT_TIME = timedelta(minutes=3)  # 3 minutes

# Fetch tweets from Twitter API
def fetch_tweets_from_twitter(country):
    query = f"{country} mental health -is:retweet lang:en"
    try:
        response = client.search_recent_tweets(query=query, max_results=100)
        return [tweet.text for tweet in response.data] if response.data else []
    except tweepy.TooManyRequests:
        print("Rate limit reached. Waiting for 5 seconds...")
        time.sleep(5)
        return fetch_tweets_from_twitter(country)

# Analyze sentiment of tweets
def analyze_sentiment(tweet):
    analysis = TextBlob(tweet).sentiment.polarity
```

```
    if analysis > 0:
        return "happy"
    elif analysis < 0:
        return "depressed"
    else:
        return "neutral"

# Save tweets to database
def save_tweets_to_db(tweets, country):
    conn = get_db_connection()
    for tweet in tweets:
        sentiment = analyze_sentiment(tweet)
        conn.execute('''INSERT INTO tweets (text, sentiment, country)
                        VALUES (?, ?, ?)''', (tweet, sentiment, country))
    conn.commit()
    conn.close()

# Fetch tweets from the database
def fetch_tweets_from_db(country):
    conn = get_db_connection()
    cursor = conn.execute('''SELECT text, sentiment FROM tweets WHERE country =
        ?''', (country,))
    tweets = cursor.fetchall()
    conn.close()
    return tweets

# Generate sentiment chart
def generate_chart(data):
```

```
    labels = list(data.keys())
    values = list(data.values())
    plt.bar(labels, values, color=['blue', 'purple', 'gray'])
    plt.xlabel("Sentiment")
    plt.ylabel("Number of Tweets")
    plt.tight_layout()

    img = io.BytesIO()
    plt.savefig(img, format="png")
    img.seek(0)
    return img

@app.route('/analyze', methods=['GET'])
def analyze():
    global last_request_time

    # Calculate time remaining for next request
    if last_request_time is not None:
        time_remaining = RATE_LIMIT_TIME - (datetime.now() - last_request_time)
        if time_remaining > timedelta(seconds=0):
            return jsonify({
                "error": "Rate limit exceeded. Try again in a few minutes.",
                "time_remaining": str(time_remaining)
            }), 429

    country = request.args.get("country", default="India", type=str)

    # Update the last request time
```

```
    last_request_time = datetime.now()

    # Try fetching tweets from the database
    stored_tweets = fetch_tweets_from_db(country)
    if stored_tweets:
        tweets = [{"text": text, "sentiment": sentiment} for text, sentiment in stored_tweets]
    else:
        # Fetch from Twitter if not stored
        tweets = fetch_tweets_from_twitter(country)
        if not tweets:
            return jsonify({"error": "No tweets found"}), 404
        save_tweets_to_db(tweets, country)

    # Perform sentiment analysis
    sentiment_results = {"happy": 0, "depressed": 0, "neutral": 0}
    for tweet in tweets:
        sentiment = tweet["sentiment"] if "sentiment" in tweet else
            analyze_sentiment(tweet["text"])
        sentiment_results[sentiment] += 1

    # Generate and send the chart
    chart = generate_chart(sentiment_results)
    return send_file(chart, mimetype="image/png")

if _name_ == '_main_':
    init_db()  # Ensure the database is created and initialized
    app.run(debug=True)
```

app.js:

```
import React, { useState, useEffect } from 'react';
import './App.css'; // Create a separate CSS file for styling

function App() {
  const [sentimentData, setSentimentData] = useState(null);
  const [loading, setLoading] = useState(false);
  const [error, setError] = useState("");
  const [timeRemaining, setTimeRemaining] = useState(0); // Time remaining for next
        request

  // Effect to update the countdown timer
  useEffect(() => {
    // Check the remaining time every second
    const timer = setInterval(() => {
      if (timeRemaining > 0) {
        setTimeRemaining(prevTime => prevTime - 1);
      }
    }, 1000);

    return () => clearInterval(timer); // Clean up the interval on unmount
  }, [timeRemaining]);

  const fetchSentiment = async () => {
    setLoading(true);
```

```
 setError("");
 try {
  const response = await fetch("http://127.0.0.1:5000/analyze?country=India");
  if (!response.ok) {
   const errorData = await response.json();
   // If time_remaining is returned by the server, set the remaining time
   if (errorData.time_remaining) {
    const [minutes, seconds] = errorData.time_remaining.split(':').map(Number);
    setTimeRemaining(minutes * 60 + seconds); // Set the remaining time in seconds
   }
   throw new Error(errorData.error || "Error fetching data");
  }

  // If the request is successful, retrieve the chart image
  const chartImage = await response.blob();
  const imageUrl = URL.createObjectURL(chartImage);
  setSentimentData(imageUrl);

  // Reset the timer to 3 minutes (180 seconds) after a successful request
  setTimeRemaining(180);
 } catch (error) {
  setError(error.message);
  setSentimentData(null);
 } finally {
  setLoading(false);
 }
};
```

```
  return (
    <div className="App">
      <h1>Sentiment Analysis for Mental Health</h1>

      {/* Error Message */}
      {error && <p style={{ color: 'red' }}>{error}</p>}

      {/* Button or Loading state */}
      {loading ? (
        <p>Loading...</p>
      ) : (
        <button onClick={fetchSentiment} disabled={timeRemaining > 0}>
          {timeRemaining > 0 ? Next Request in ${timeRemaining}s : 'Fetch Sentiment'}
        </button>
      )}

      {/* Displaying the sentiment chart image */}
      {sentimentData && <img src={sentimentData} alt="Sentiment Chart" />}
    </div>
  );
}

export default App;
```

REFERENCES

1. Evaluating Mental Health Sentitement on Social Media During COVID-19-[KHANDELWAL,V]-2024

2. Enhancing Mental Health Awarness Throuogh Twitter Analysis – [CHATTERJEE, R.,]-2023

3. Sentitement Analysis on Mental Health Campaign Twetts from 2017 to 2023 –[Z.Hama,M]-2023

4. Sentitement About Mental Health on Twitter—Before and During the COVID-19 PANDEMIC-[FELIX BEIERL]-2023

5. Multi-Class Depression Detection Through Tweets Using Artificial Intelligence [Zhao, Y., Zhou, H., & Wang, X.] (2023).

6. Data Fusion Techniques for Depression Detection in Tweets – [Jamal, I., & Aribisala, F.] (2023).

7. Fine-Tuned BERT Models for Sentiment Prediction in Mental Health Tweets [Nusrat, A., & Islam, M.] (2023).

8. A Survey of Sentiment Analysis: Approaches, Datasets, and Future Research [Kian Long Tan, Chin Poo Lee, Kian Ming Lim] (2023)

9. VADER Sentiment Analysis on Mental Health Campaign Tweets – [Hama, Z., et al.] (2023).

10. Sentiment Analysis on Mental Health Discourse Using Machine LearningApproaches – [Singh, A., & Rajan, P.] (2023).