**MCA 202- Software Engineering Lab**

**WHATSINSIGHT - WhatsApp Chat Analyser**

**MCA202 Software Engineering Project Report**

**Mid-Semester Evaluation**

**Submitted by:**

**(2024010106) Srishti**

**(2024010111) Sunishtha Singh**

**(2024010112) Supreet Kaur**

**(2024010118) Vinayak Chhabra**

**MCA First Year**

**Team ID: 1**

**Submitted to:**

**Dr. Harkiran Kaur**

**(Assistant Professor)**

****

**Computer Science and Engineering Department**

**TIET, Patiala**

**March 2025**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Assignment** | **Page No.** |
| **1.** | **Planning Phase** | 3 |
| 1.1 | i. Project Write Up | 3 |
| **2.** | **Analysis Phase** | 4 |
| 2.1 | i. Use Case Template | 4 |
| 2.2 | ii. Use-Case diagram | 6 |
| 2.3 | iii. Data Flow Diagrams - Level 0, Level 1 | 7 |
| 2.4 | iv. Software Requirement Specification (SRS) in IEEE Format | 8 |
| **3.** | **Design Phase** | 17 |
| 3.1 | i. Class Diagram | 17 |
| 3.2 | ii. Module Structure Chart | 18 |
| **4.** | **Project Progress (Screenshots)** | 18 |

1. **PLANNING PHASE**

**1.1 Project Write Up**

**Need for the Project**

In today’s digital era, WhatsApp is one of the most widely used messaging platforms, generating vast amounts of chat data that hold valuable insights into communication patterns, user behavior, and sentiments. However, manually analyzing such data is impractical.

**This project aims to develop an automated solution to extract meaningful insights from WhatsApp conversations, enhancing understanding of messaging trends and engagement using sentiment analysis.**

Additionally, the rise of online harassment, cyberbullying, and inappropriate content in messaging apps is a growing concern. To promote safer communication, the project includes a **detective module** that detects and flags negative, abusive, or threatening messages, fostering a more secure and respectful messaging environment.

**Functional Requirements/Features**

* **Chat Statistics** – Message count, word count, links, media shared, and relationship type.
* **Timeline Analysis** – Daily, weekly, and monthly chat frequency trends.
* **Activity Map** – Identifies peak messaging hours and days.
* **Most Busy Users** – Displays user activity in a bar chart.
* **Word Cloud** – Highlights frequently used phrases.
* **Red Flag Detection** – Flags messages containing abusive, illegal, or insensitive words.
* **Detective Mode (Extra Feature)** – Detects hate speech, showing the source user in a bar chart.

**Quality Attributes**

* **Usability** – Intuitive user interface with easy navigation, ensuring accessibility for both technical and non-technical users.
* **Security** – Authorization levels for **Detective Mode** and **User Mode** to control access.
* **Ease of Implementation** – Designed for straightforward deployment with minimal technical complexity.

1. **ANALYSIS PHASE**

**2.1 Use Case Template**

|  |  |
| --- | --- |
| 1. Use case title | Login |
| 1. Abbreviation | Login |
| 1. Use Case Id | 1 |
| 1. Actors | Standard, Investigator |
| 1. Description – Pre-existing users can log in with their credentials and access their dashboard – based on their level of access | |
| 5.1 Pre-conditions – User should be registered priorly | |
| * 1. Task Sequence – 1. Spaces to enter username, password and login button will be displayed on screen   2. User to enter their credentials  3. On clicking login, user will be authenticated and taken to their respective dashboard based on their level of access | |
| 1. Modification History: | |
| 1. Author: Vinayak, Sunishtha | |

2.1.1 Login

2.1.2 Sign Up

|  |  |
| --- | --- |
| 1. Use case title | Sign up |
| 1. Abbreviation | Sign up |
| 1. Use Case Id | 2 |
| 1. Actors | Standard, Investigator |
| 1. Description – New user will be able to provide their details and get credentials to use the features of the tool | |
| * 1. Pre-conditions – User should have existing email id, phone no. | |
| 5.2 Task Sequence – 1. Sign up button will be displayed on the login screen  2. User can provide their email, phone number details and create a password  3. New credentials will be generated for the user | |
| 1. Modification History: | |
| 1. Author: Vinayak, Sunishtha | |

2.1.3 Import Chat

|  |  |
| --- | --- |
| 1. Use case title | Import/upload chat |
| 1. Abbreviation | Import chat |
| 1. Use Case Id | 3 |
| 1. Actors | Standard, Investigator |
| 1. Description – Allows logged in user to upload .txt file of the exported WhatsApp chat | |
| 5.1 Pre-conditions – User should be logged in | |
| * 1. Task Sequence – 1. Upload button will be displayed on screen   2. Browse from PC/drive and upload .txt file  3. On clicking upload, the whole .txt file will be uploaded – ready for the model to analyse | |
| 1. Modification History: | |
| 1. Author: Vinayak, Sunishtha | |

2.1.4 Chat Analysis

|  |  |
| --- | --- |
| 1. Use case title | Chat Analysis |
| 1. Abbreviation | Chat Analysis |
| 1. Use Case Id | 4 |
| 1. Actors | Standard, Investigator |
| 1. Description – Allows user to start analysis of the exported WhatsApp chat | |
| * 1. Pre-conditions – User should be logged in | |
| 5.2 Task Sequence – 1. Start Analysis button displayed on screen  2.Once clicked, model will start analysing the chat and display overall statistics once done | |
| 1. Modification History: | |
| 1. Author: Vinayak, Sunishtha | |

2.1.5 Timeline Analysis

|  |  |
| --- | --- |
| 1. Use case title | Timeline Analysis |
| 1. Abbreviation | Timeline Analysis |
| 1. Use Case Id | 5 |
| 1. Actors | Standard, Investigator |
| 1. Description – User will be able to see daily, weekly and monthly activity timeline of included chat parties | |
| * 1. Pre-conditions – User should be logged in | |
| 5.2 Task Sequence – 1. Timeline Analysis displayed on sidebar navigator  2. Display analysed data on the basis of activity timeline – visualised in the form of heatmaps and line graphs | |
| 1. Modification History: | |
| 1. Author: Vinayak, Sunishtha | |

2.1.6 Most Busy Users

|  |  |
| --- | --- |
| 1. Use case title | Most Busy Users Analysis |
| 1. Abbreviation | Most Busy Users |
| 1. Use Case Id | 6 |
| 1. Actors | Standard, Investigator |
| 1. Description – User will be able to display the most active users in a conversation. | |
| * 1. Pre-conditions – User should be logged in | |
| 5.2 Task Sequence – 1. Most Busy Users Analysis option displayed on sidebar navigator  2. Display analysed data on the basis of frequency of a user’s messages – visualized in the form of bar graph | |
| 1. Modification History: | |
| 1. Author: Vinayak, Sunishtha | |

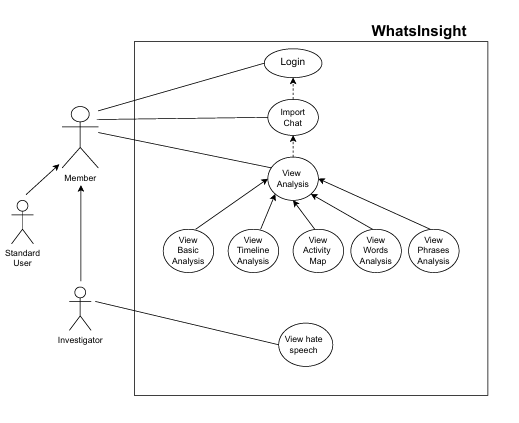
2.1.7 Hate Speech Analysis

|  |  |
| --- | --- |
| 1. Use case title | Hate Speech Analysis |
| 1. Abbreviation | Hate Speech Analysis |
| 1. Use Case Id | 7 |
| 1. Actors | Investigator |
| 1. Description – Investigator | |
| * 1. Pre-conditions – User should be logged in and have access to Detective mode | |
| 5.2 Task Sequence – 1. User clicks on Detective Mode option  2. Permitted user can view red flag detected and sentiment of extreme messages of particular chat members | |
| 1. Modification History: | |
| 1. Author: Vinayak, Sunishtha | |

2.1.8 Most Abusive User

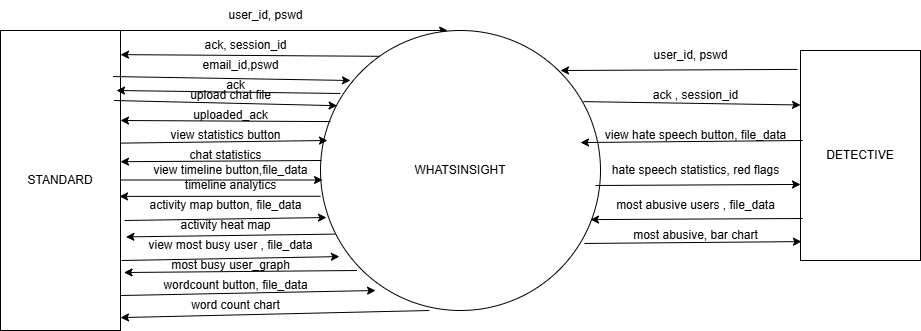
|  |  |
| --- | --- |
| 1. Use case title | Most Abusive User Analysis |
| 1. Abbreviation | Most Abusive User |
| 1. Use Case Id | 8 |
| 1. Actors | Investigator |
| 1. Description – User will be able to see bar chart of source users of most frequent abusive/hate speech. | |
| * 1. Pre-conditions – User should be logged in and have access to Detective mode | |
| 5.2 Task Sequence – 1. User clicks on Detective Mode option  2. Permitted user can view red flag detected and sentiment of extreme messages of particular chat members | |
| 1. Modification History: | |
| 1. Author: Vinayak, Sunishtha | |

**2.2 Use-Case Diagram**

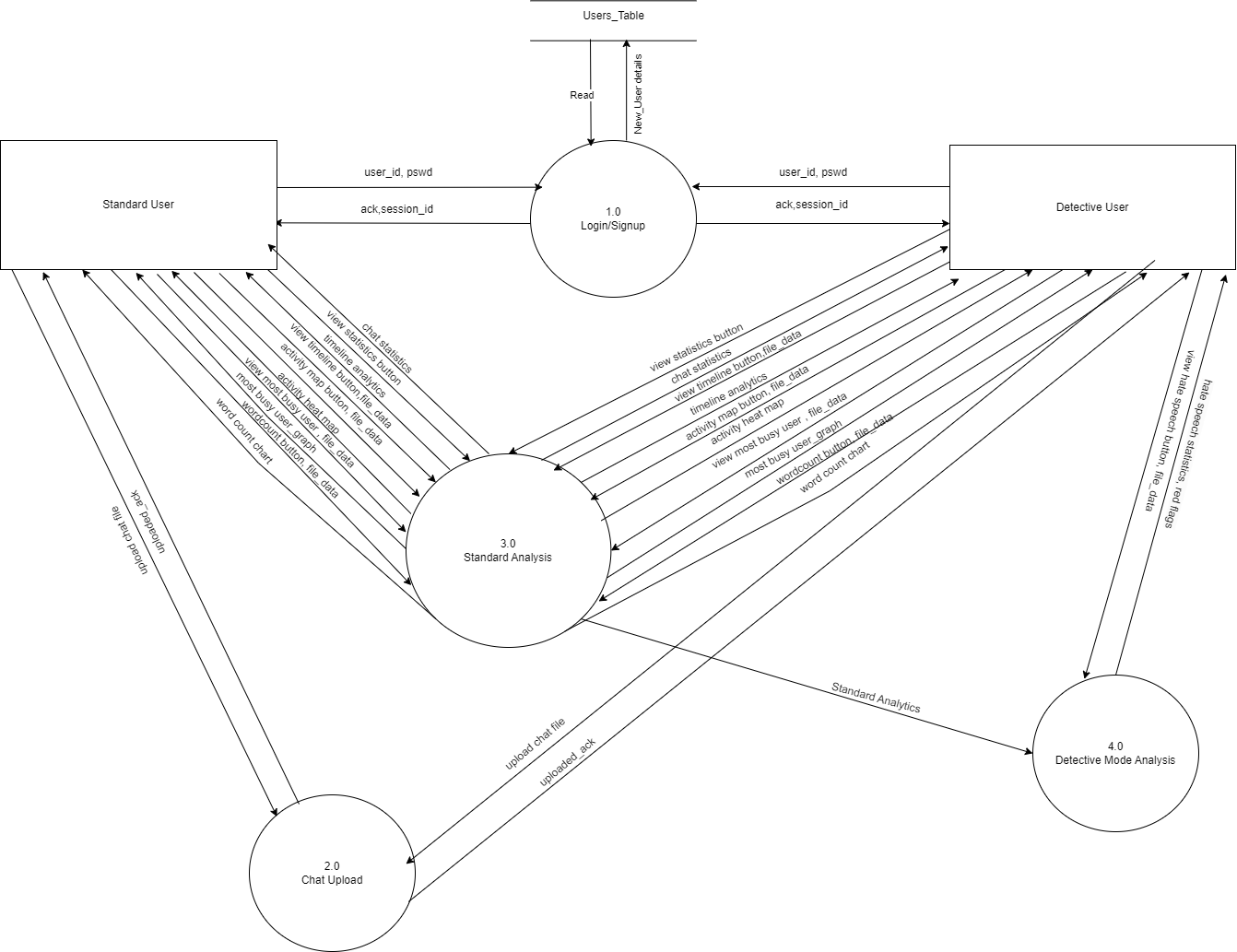


**2.3 Data Flow Diagrams**

Level 0:



Level 1:



**2.4 Software Requirement Specification (SRS)**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Topic** | **Page No.** |
| 1 | Introduction | 10 |
| 1.1 | Purpose of this Document | 10 |
| 1.2 | Scope of the development project | 10 |
| 1.3 | Definitions, Abbreviations and Acronyms | 11 |
| 1.4 | References | 11 |
| 1.5 | Overview | 11 |
| 2 | Overall Description | 12 |
| 2.1 | Product perspective | 12 |
| 2.2 | Product Function | 12 |
| 2.3 | User Characteristics | 12 |
| 2.4 | General Constraints | 12 |
| 2.5 | Assumptions and Dependencies | 12 |
| 3 | Specific Requirement | 13 |
| 3.1 | External Interface requirements | 13 |
| 3.2 | Detailed Description of Functional Requirements | 13 |
| 3.2.1 | Chat Statistics | 13 |
| 3.2.2 | Timeline Analysis | 14 |
| 3.2.3 | Activity Map | 14 |
| 3.2.4 | Most Busy Users | 14 |
| 3.2.5 | Word Cloud | 14 |
| 3.2.6 | Red Flag Detection | 15 |
| 3.2.7 | Detective Mode | 15 |
| 3.3 | Performance Requirements | 15 |
| 3.4 | Logical Database Requirement | 16 |
| 3.5 | Quality Attributes | 16 |
| 4 | Change History | 16 |
| 5 | Document Approves | 17 |

1. **INTRODUCTION**
   1. **Purpose of this Document**

This document provides a detailed overview of **WhatsInsight** – our software product, which is designed to analyze WhatsApp chats, its parameters and outcomes. It also includes the target audience of the product, the user interface specifications, functional requirements detail along with the quality attributes inculcated in the product.

* 1. **Scope of the Development Project**

**WhatsInsight** is a fun yet useful app designed to extract meaningful insights from WhatsApp chat exports. It will be able to analyze the communication patterns, word usage, frequency of interaction along with performing overall sentiments analysis. It also includes a feature that automatically flags the chats of individuals indulging in abusive or illegal language use – this enabling the user or any person of authority investigating the chat to be able to identify use of hate speech.

This analysis will be done using an ML model along with Natural Language Processing to detect sentiment and inappropriate content between involved parties.

The software must be able to perform following operation

* **Identify and authenticate** - Simple login functionality
* **Sign up new user** -Allow new users to sign up to the tool and create their credentials
* **Role based access** - Distinguish user type on basis of their login credentials and display dashboard according to their level of access
* **File Upload -** Allow user to upload exported WhatsApp chat files in .txt format
* **Overall Chat Analysis** - Analyze upload chat and provide chat statistics once analysis is completed, including message and word count, media and links shared, overall sentiment and possible relationship.
* **Timeline Analysis** – Display line graph and activity heatmaps of daily, weekly and monthly activity timeline between individual and group users
* **Word Cloud** – Pictographically represent most used words or phrases
* **Hate Speech Detection & Representation** - Detect and flag abusive, insensitive, or illegal words and create a bar chart of sources of interactions, specifically in Detective mode
* **Ensure data privacy** – no chat should be stored permanently
  1. **Definitions, Abbreviations and Acronyms**

Table below gives explanation of the most commonly used terms and abbreviations in this SRS document

|  |  |
| --- | --- |
| WhatsApp | Social platform allowing users with phone numbers to exchange conversation via text, image, voice etc. |
| ML Model | Machine Learning Model built that performs complete analysis of exported chats |
| NLP | Natural Language Processing – technique used to analyze natural language sentiment in ML |
| Detective Mode | Highly authorized mode – meant for officials investigating WhatsApp chat patterns for possible criminal case |

* 1. **References**

WhatsApp - <https://en.wikipedia.org/wiki/WhatsApp>

ML Model - <https://www.databricks.com/glossary/machine-learning-models#:~:text=A%20machine%20learning%20model%20is,sentences%20or%20combinations%20of%20words>.

NLP - <https://aws.amazon.com/what-is/nlp/#:~:text=Natural%20language%20processing%20(NLP)%20combines,with%20computers%20and%20software%20tools>.

* 1. **Overview**

The remaining sections of this document provide a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product. General description of the project is discussed in section 2 of this document. Section 2 gives the functional requirements, data requirements and constraints and assumptions made while designing the multi-utility system. It also gives the user viewpoint of product use. Section 3 gives the specific requirements of the product. Section 3.0 also discusses the external interface requirements and gives detailed description of functional requirements.

1. **OVERALL DESCRIPTION**
   1. **Product Perspective**

The **WhatsInsight** is a **standalone web application** with a **server-side NLP model** for chat analysis. It does **not require real-time API access** to WhatsApp but works with manually exported **.txt chat files**.

* 1. **Product Functions (Functional Requirements)**

The product will be able to perform the following functions:

* Identify and authenticate user using their login credentials. Additionally, allow new users to sign up to the tool.
* Distinguish type of user mode – standard or detective and accordingly adjust functionalities, ensuring safe access.
* Allow user to upload exported WhatsApp chat files in .txt format
* Analyze WhatsApp exported chat files (.txt format).
* Provide chat statistics i.e. message count, word count, and media shared.
* Generate visual insights word clouds, activity heatmaps, and bar charts.
* Detect and flag abusive, insensitive, or illegal words.
* Offer a **detective mode** to track hate speech sources.
* Ensure data privacy – no chat should be stored permanently
  1. **User Characteristics**

|  |  |  |
| --- | --- | --- |
| **User Type** | **Description** | **Permissions** |
| Standard User mode | General Users analyzing personal chats | Upload chats, view analytics |
| Detective User mode | User analyzing flagged messages | Access flagged messages and respective senders |

* 1. **Constraints**
* Only text-based chats are supported (no media processing).
* No direct WhatsApp integration (manual export required).
* Processing large chats (100k+ messages) world require xserver-side optimization.
  1. **Assumptions and Dependencies**
* Users must manually export chats from WhatsApp
* The NLP model must be trained to avoid false positives
* The system must comply with data privacy regulations

1. **SPECIFIC REQUIREMENTS**
   1. **External Interface Requirements**

The following list represents the external interface requirements:

* **User-Friendly Dashboard** – The system should have a simple, structured and clean UI design with easy to operate features, a sidebar navigation and easily accessible drop-downs/buttons for different chat statistic options. There should not be long horizontal or vertical scroll range.
* **File Upload & Processing** – To display chat file upload, a small representation in the form of a progress indicator should be present, displaying status of file upload.
* **Chat Analysis** – A button allowing user to start analysis once upload is complete. A buffer/processing animation indicating processing of chats to be included.
* **Data Visualization** – The system shall generate bar charts (most active users), line charts (chat frequency), and heat maps (peak chat hours) to help users interpret chat data visually.
* **Word Cloud Generation** – A word cloud shall be created from the chat text to highlight the most frequently used words and phrases in a visually engaging manner.
* **Flagged Message Detection** – The system shall detect messages containing abusive, illegal, or insensitive words and display them in a structured table with color-coded alerts**.**
* **Detective Mode (Additional Feature)** – Detective mode users shall have an option to enable Detective Mode, which will analyse chat data for hate speech detection and present results in a bar chart identifying source users.
* **Colours & Contrast** – The UI should not have extreme colour scheme but should highlight important portions, especially red flag for hate speech messages.
* **Export & Reports** – Users shall be able to download chat analysis reports in CSV format, summarizing statistics, flagged messages, and insights for further review.
  1. **Detailed Descriptions of Functional Requirements**

**3.2.1 Chat Statistics:**

|  |  |
| --- | --- |
| **Functionality** | **Chat Statistics** |
| **Purpose** | Provide overall chat summary including message and word count, links, media, and relationship type. |
| **Inputs** | WhatsApp exported chat file (.txt format). |
| **Processing** | Extract message count, word count, media, and links. Identify relationship type based on frequency of interaction. |
| **Outputs** | Total messages, word count, number of links/media, relationship type classification. |

**3.2.2 Timeline Analysis:**

|  |  |
| --- | --- |
| **Functionality** | **Timeline Analysis** |
| **Purpose** | Show chat frequency trends over time. |
| **Inputs** | Timestamped chat messages. |
| **Processing** | Aggregate message timestamps into daily, weekly, and monthly trends. |
| **Outputs** | Line graph displaying chat activity over different time periods. |

**3.2.3 Activity Map:**

|  |  |
| --- | --- |
| **Functionality** | **Activity Map** |
| **Purpose** | Identify peak messaging hours and active days. |
| **Inputs** | Timestamped chat messages. |
| **Processing** | Analyse message frequency by hour and day. |
| **Outputs** | Heatmap showing peak chat activity by time of day and day of the week. |

**3.2.4 Most Busy Users:**

|  |  |
| --- | --- |
| **Functionality** | **Most Busy Users** |
| **Purpose** | Display the most active users in a conversation. |
| **Inputs** | Sender details extracted from chat messages. |
| **Processing** | Count messages per user and rank them by activity. |
| **Outputs** | Bar chart ranking users based on message count. |

**3.2.5 Word Cloud:**

|  |  |
| --- | --- |
| **Functionality** | **Word Cloud** |
| **Purpose** | Highlight frequently used words and phrases. |
| **Inputs** | Chat messages (excluding common words). |
| **Processing** | Tokenize text, count word frequency, and generate a word cloud. |
| **Outputs** | Word cloud visualization displaying top words and phrases. |

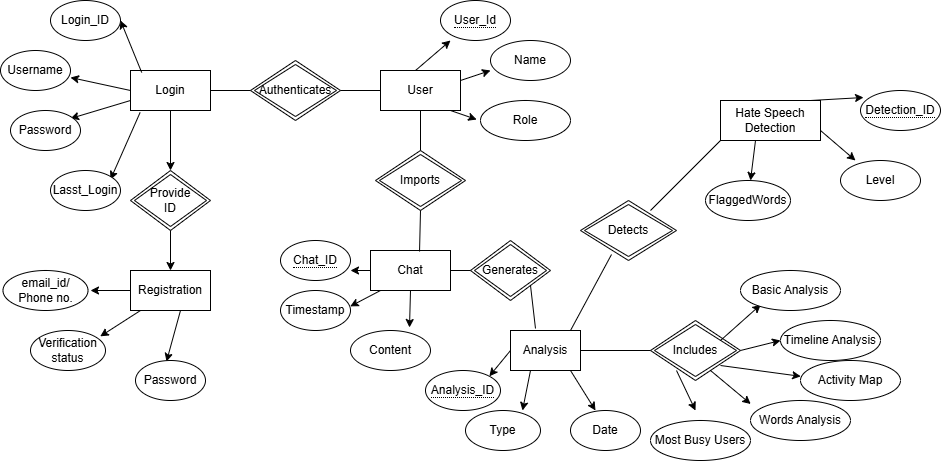
**3.2.6 Red Flag Detection:**

|  |  |
| --- | --- |
| **Functionality** | **Red Flag Detection** |
| **Purpose** | Identify and flag messages containing abusive, illegal, or insensitive words. |
| **Inputs** | Parsed chat text. |
| **Processing** | Compare messages with a predefined list of offensive words using NLP models. |
| **Outputs** | List of flagged messages and notification alerts. |

**3.2.7 Detective Mode:**

|  |  |
| --- | --- |
| **Functionality** | **Detective Mode (Extra Feature)** |
| **Purpose** | Detect hate speech and identify the source user. |
| **Inputs** | Parsed chat text. |
| **Processing** | Apply NLP-based hate speech detection algorithms. Generate user-wise flagged message count. |
| **Outputs** | List of flagged hate speech messages with a bar chart displaying source users. |

* 1. **Performance Requirement**
* **Accurate Parsing and Extraction**: The app should be able to accurately interpret and extract key elements from WhatsApp chat data (messages, media, timestamps, etc.) and provide accurate analysis, including sentiment analysis, keyword extraction, or user interaction metrics.
* **Scalability and Flexibility**: The app should be able to handle varying amounts of chat data without performance issues or any interruptions, from small to bulky chat data, and allow for flexible analysis (e.g., single conversation vs. multiple group chats).
* **Data Privacy and Security**: The app should handle chat data securely, ensuring that user privacy is maintained. This includes encryption of data during storage and transmission and compliance with relevant privacy laws and standards (e.g., GDPR).
* **Robust Search and Filter Capabilities**: The app should allow users to quickly search for specific messages, dates, keywords, or contacts within large chat histories. Filtering capabilities should be fast and accurate, enabling customized reports.
* **User-Friendly Interface:** The app should be easy to use, with a clean and intuitive interface that lets users explore their chat data and uncover insights effortlessly—no complicated steps or learning curve required.
* **Cross-Platform Compatibility**: The app should work seamlessly on Android, iOS, and web platforms, delivering fast and consistent performance so users enjoy the same smooth experience no matter what device they're using.
  1. **Logical Database Requirement**

****

**3.5 Quality Attributes**

* **Usability** – Intuitive user interface with easy navigation, ensuring accessibility for both technical and non-technical users.
* **Security** – Authorization levels for **Detective Mode** and **User Mode** to control access.
* **Ease of Implementation** – Designed for straightforward deployment with minimal technical complexity.

**3.6 Other Requirements**

None at this time

1. **CHANGE HISTORY**

|  |  |
| --- | --- |
| **07111218** | Version 1.0 – Initial Release |
|  |  |

1. **DOCUMENTS APPOROVERS**

SRS for CSC based Multi-Utility System (including Access Control and Attendance Monitoring) approved by:

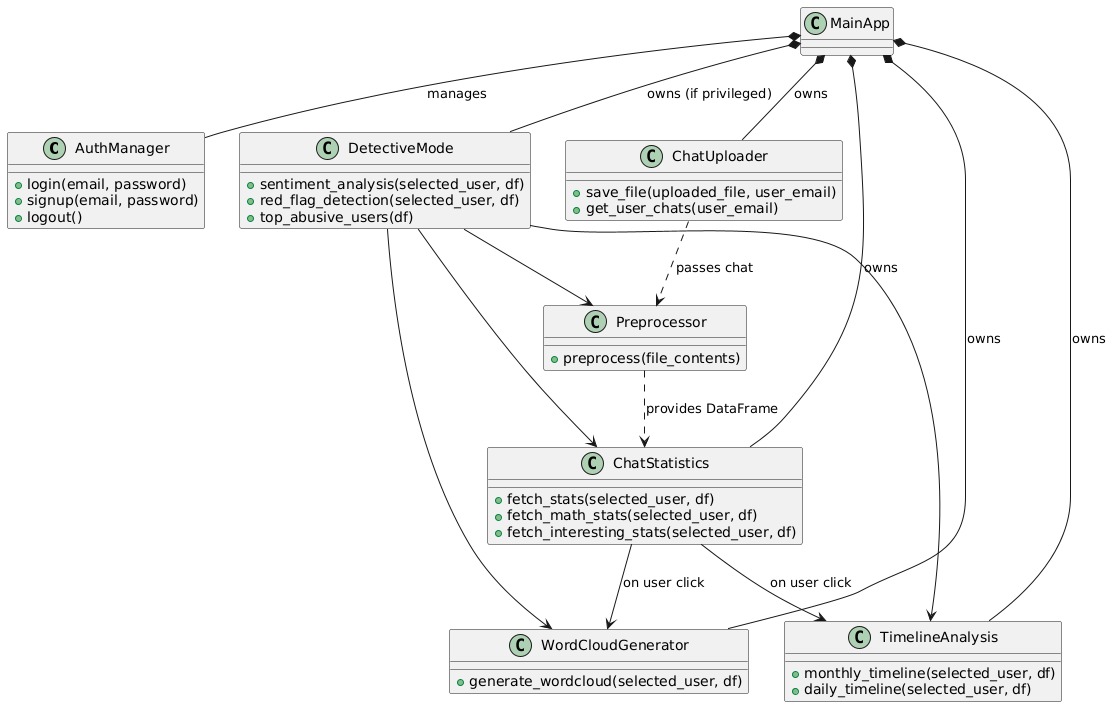
Name: Dr. Harkiran Kaur

Designation: Assistant Professor

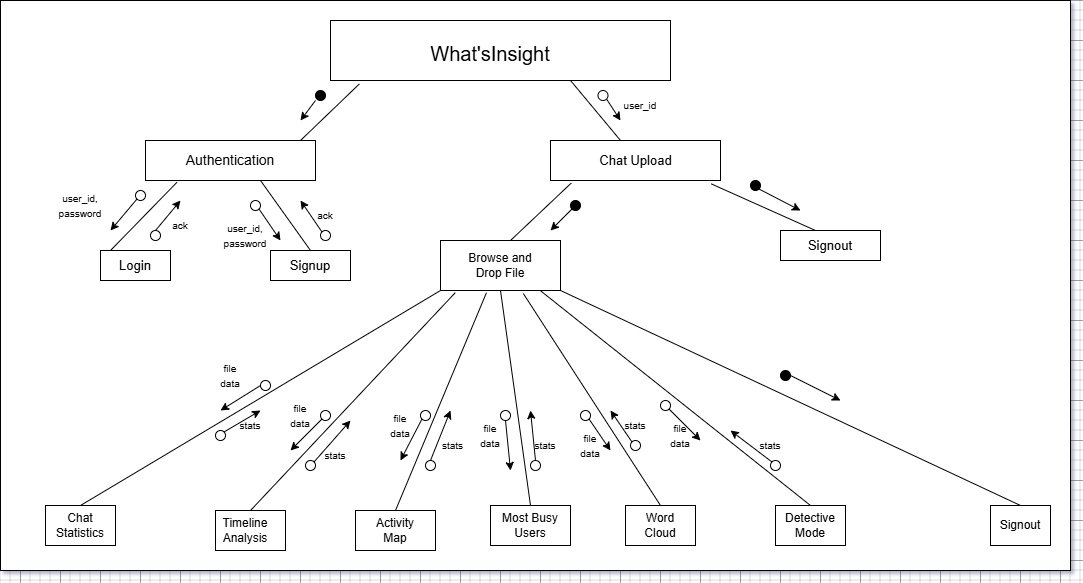
Date: 05-02-2025

1. **DESIGN PHASE**

i. Class Diagram



ii. Module Structure Chart



1. **PROJECT PROGRESS**

