iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: V Month of publication: May 2023

DOI: https://doi.org/10.22214/ijraset.2023.51410

www.ijraset.com

Call: ✆08813907089 | E-mail ID: ijraset@gmail.com

Chat Application with Profanity Check

Ayush Ahuja[1], Gautam Chaurasia[2], Aakash Gupta[3], Dr. Sheetal Rathi[4]
[1, 2, 3]UG Student, [4]Prof. Dean Academy Department of Computer Engineering, Thakur College of Engineering & Technology, Mumbai, India

Abstract: The use of chat applications has increased to a large extent because it provides users with the facility to interact with their family and friends instantly. But these chat applications are being used for carrying out some unethical practices like cyber-bullying, online harassment, etc and there is a need to find a way to curb this issue. This paper aims to give a clear insight into solutions currently being employed or suggested to overcome this problem and to identify the drawbacks associated with these solutions on a large scale. It also proposes a working model for chat apps with profanity detection based on the gaps identified which can detect foul language and thus it can significantly reduce these unethical practices prevalent today in the world of social media.

Keywords: Profanity, Foul Language, Chat Application, Video Conference, Media Sharing, Natural Language Processing

I. INTRODUCTION

Since the beginning of time, communication has been a way of sharing thoughts, emotions, data, and information with others. With the advent of languages, people started expressing themselves both verbally and through writing. Now, by connecting to the internet, computers anywhere in the globe may exchange data. Different computer technologies have improved data transport and communication.

Virtual interactions are becoming more and more popular as a means of social communication. The ability of chat applications to replicate the feel of a real-time conversation virtually has helped them become increasingly popular across a variety of applications, including gaming group messaging in a live chat, e-learning, team communication using chat rooms, and file sharing among coworkers. But while planning how we want to develop our web app, it's important to keep our consumers' experience in mind. What key attributes and capabilities are required to create a web chat application that keeps users interested.

II. MOTIVATION

With the internet being used the most in today’s environment, everyone has sometime or the other experienced cyber bullying or online hate. This has become a problem that needs to be addressed with a huge amount of concern. No human being should be subjected to unwanted abuse, which has been quite prevalent in modern society. Children, who are naïve and innocuous should not be burdened by trauma or fear instead of being instilled with curiosity and aspirations. Every human being has the right to be spoken with respect and courtesy and there is a need to ensure that this right is being accessed by everyone on the internet where the exploitation of the freedom of speech has become a new norm. This step would potentially cut off cybercrimes and make the internet much healthier.

Apart from a healthier internet, there’s also the need for a convenient one stop solution for all our communication needs. Currently existing products don’t offer every feature desired by a modern user all in one package, for media sharing and online chatting a product/ service is being used and to make live, one to one or group video calls another service. The video conferencing solutions lack in providing robust and reliable media sharing from within their products/ services. This continuous transition between different products for using different sets of features is a cause for great inconvenience.

III. LITERATURE SURVEY

The rise of profanity in recent years has been alarming. This has led to the compelling need for a reliable method to control such profanity.

Reading through several study studies reveals that an ideal integration of Natural Language Processing and Machine Learning would help create the necessary specific model.

[1] This research paper aims to maintain security, however it does not focus on the web application’s performance , which results in a longer response time for the user. [2]The approach to the problem in this paper is quite generalized as no key differentiating feature has been introduced to encourage large-scale adoptability of the product and is limited to a few selected subset of users. [3]

A drawback to the given solution in this paper is that it is limited to only within private networks. It introduces a very peculiar problem by limiting the use to private networks.

[4] The idea proposed in this paper doesn't allow the user to send the message if the model detects the context of the message is negative, which means it reads the message even before it is sent at the sender's side. [5] This paper focuses on developing a filter for chat applications where each message is checked for toxicity using Deep Learning and Natural Language Processing, where the focus is on understanding the context of each message. [6] This paper introduced the working and implementation of encrypted chat applications. It describes how the messages are encrypted on device before moving to server using various cryptographic algorithms like DES, AES and RSA. Computer vision techniques are used for encrypting image

IV. METHODOLOGY

The entire system is a web based application and can be broken down into three different modules, the chat application, the video conferencing module and the profanity check model.

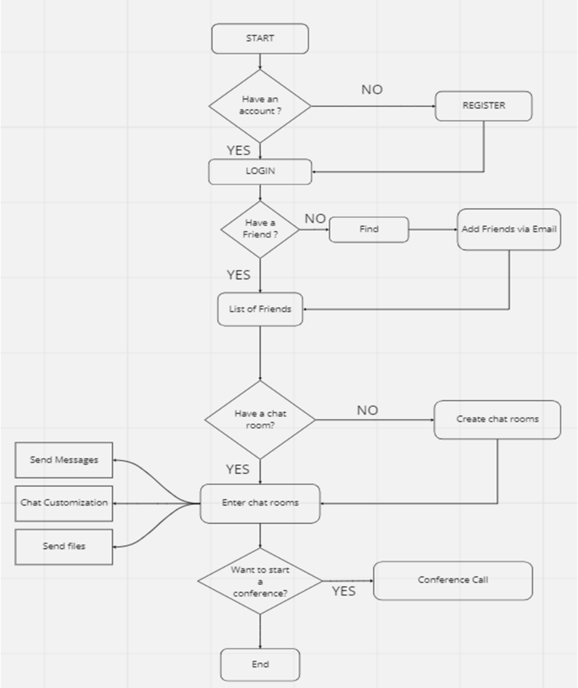


Fig 1. Flowchart of System

1) The login page will be displayed before the chat page when a user accesses the chat page. User login information is necessary. The user must first register if he/she doesn't have an account.
2) Users must have a friend after logging in before starting a chat. As a result, if a user doesn't already have a friend, they can add a friend directly from the email.
3) The user can start a chat immediately, if they have contacts in their friend list.
4) The application will then check to see if the user is already in a chatroom with the specified user . If the chat room is not present then the user will enter it to begin a chat session .
5) Finally, data will be sent and received.

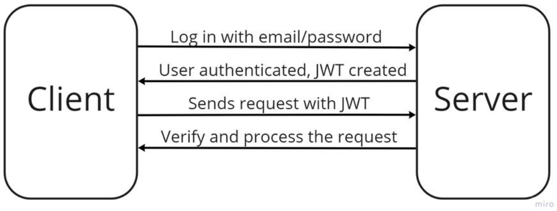

Client
Log in with email/password
User authenticated, JWT created
Sends request with JWT
Verify and process the request
Server
miro

Fig 2. LogIn Process

a) Conferencing Module:

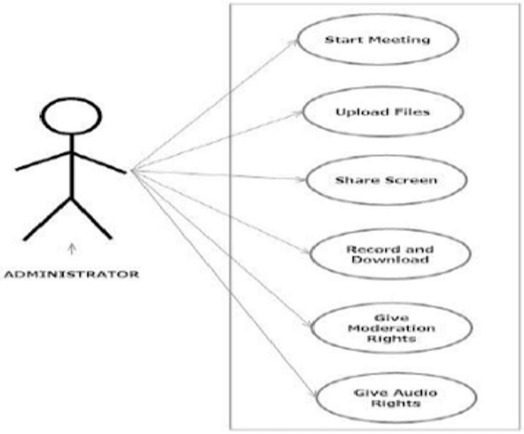


Fig 3. Use Case Diagram for conference - Admin

The administrator gains access to every part of the model and are permitted to make changes when necessary, add and remove moderation rights, allow or deny the users to draw on the white board, allow or deny the users screen sharing or recording, allow or deny audio rights to users, give exclusive audio right to self and upload files

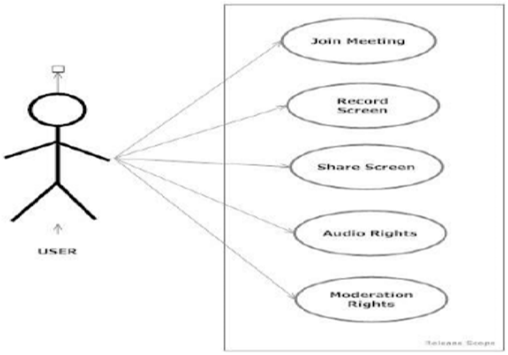


Fig 4. Use Case Diagram for conference - User

The user is only permitted to join the meeting when invited via an email, record screen, share screen, use the white board, and audio when given permission by the administrator.
Server Side Implementation:
Server-side has relied on Node JS and MongoDB databases. Node JS is fast, capable of handling a large number of simultaneous connections with high throughput, which is equivalent to high scalability. MongoDB and Node JS have often been used together because of their using JSON so no need to spend time transforming the data between them making it easy to deal with each other. In addition, MongoDB provides TLS that makes a secure connection . To perform a client request passes through several steps that are:
Step 1: Initially, must run the MongoDB connection then run the Node JS from Command Prompt. At this stage, the server is ready to receive the client's request.
Step 2: When the client sends a request, the server receives the HTTP request in JSON format. The request then parsed.
Step 3: The HTTP request is compared with the base path if it is matched, it is handed to the Express framework.
Step 4: The Express receives the HTTP request and routes it to the specific endpoint that matched it. In case of not matching with any of the routes will display errors in Command Prompt. Otherwise, it will be forwarded to the controller which handles the required function. Step 5: Make a request to MongoDB database by mongoose for processing function.
Step 6: When the data is fetched from MongoDB database and the required operations are done, Node JS receives the response then sends to the client

b) Profanity Check Module:

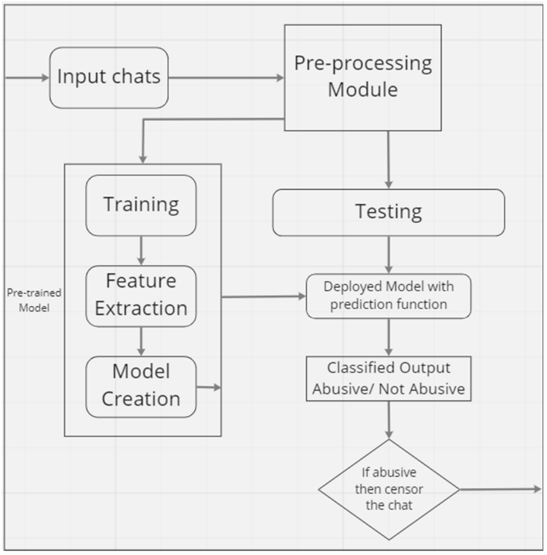

Input chats
Pre-processing Module
Training
Feature Extraction
Model Creation
Pre-trained Model
Testing
Deployed Model with prediction function
Classified Output Abusive/ Not Abusive
If abusive then censor the chat

Fig 5. Profanity Check Model

Whenever a message is sent, it will pass on through the profanity check module. This module will pre-process the chat i.e removal of punctuations, numbers, conversion of text to lower case and NLP concepts of removing stop words, stemming, tokenization, named entity recognition and parts of speech tagging, it gives keywords from the chat. These keywords will be passed on through our trained model to decide the sentiment of the chat (abusive or normal).

V. CONCLUSION

We are building one application that will incorporate all the features for chatting as well as hosting meetings which previous products failed to provide.

From our research we can conclude that, the performance parameter can be improved by using the React framework, the concept of virtual space improves the performance rate by only altering the space created for that specified request instead of making a call for the altering of the whole work area. So MERN stack is the best optimal technology to be used based on performance.

In this chat application the user will be able to type in their own language after selecting the language of their comfort, then the text will be recognized and translated to the language opted by the end user.

So, we will incorporate multi-headed machine learning models that are capable of classifying messages sent by users into varying degrees of toxicity. This application will track the use of offensive and vulgar language on chats on the platform in order to prevent users from becoming disrespectful which may play a role in changing how users interact.

VI. ACKNOWLEDGEMENT

We the members of Group A08 would like to express our special gratitude to our institute, Thakur College of Engineering and Technology, our Principal sir - Dr. B. K. Mishra, Head of Department - Mrs. Harshali Patil, and our mentor Dr. Sheetal Rathi, for giving us this opportunity to work on solving the identified problem statement. We are developing a WebApplication named “PROTALK” by using various programming tools, Frameworks in multiple languages along with the assistance of Kaggle for the dataset. This has greatly helped us in expanding our knowledge. The project will be completed within the given scheduled time under the guidance of our mentor. We would also like to expand our deepest gratitude to all those who have directly and indirectly guided us in developing this project..

REFERENCES

[1] “Multi-User Chat Application” - R. Gayathri, C. Kalieswari, International Research Journal of Engineering and Technology (IRJET) International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958 (Online), Volume-9 Issue-5, June 2020

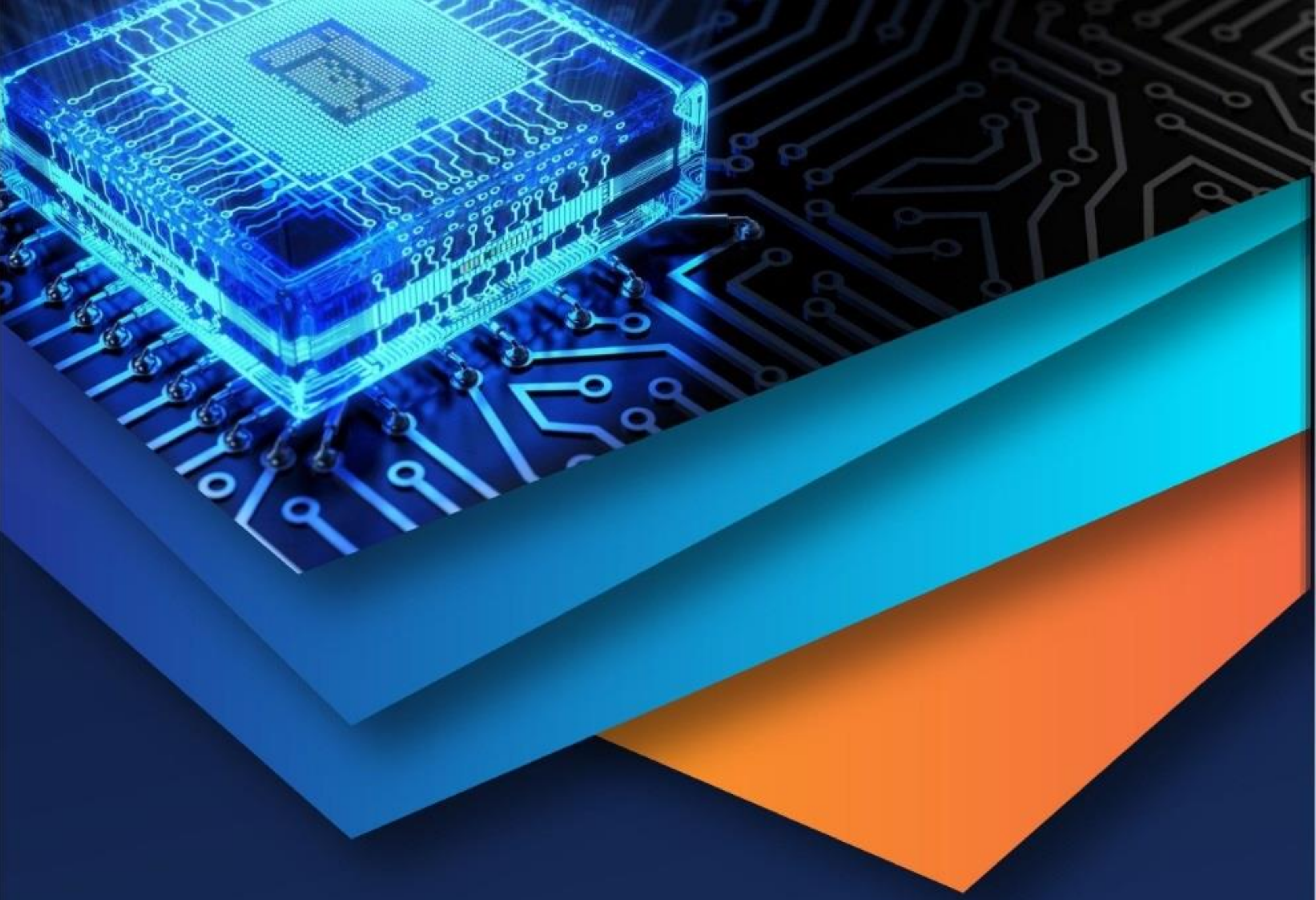
[2] “Design and Implementation of Web Based Real Time Chat Interfacing Server” - Diotra Henriyan, Devie Pratama Subiyanti, Rizki Fauzian, Dian Anggraini, M. Vicky Ghani Aziz, Ary Setijadi Prihatmanto, 2016 IEEE 6th International Conference on System Engineering and Technology (ICSET) October 3-4, 2016 Bandung – Indonesia

[3] "Design and Implementation of a Virtual Machine Video Conferencing Application" - Adeyinka A. Adewale, Samuel N. John, Dike U. Ike, International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 3 Issue3 March, 2014 Page No. 5054-5063

[4] "Professional Chat Application based on Natural Language Processing" - Karthick S, R John Victor, Manikandan S, Bhargavi Goswami

[5] "Profanity Filter and Safe Chat Application using Deep Learning" - Sanjana Kumar, Srikrishna Veturi, Varun Sreedhar, International Research Journal of Engineering and Technology (IRJET) Volume: 08 Issue: 07 | July 2021

[6] Sabah, Noor, Jamal M. Kadhim, and Ban N. Dhannoon. "Developing an End-to-End Secure Chat Application." IJCSNS 17.11 (2017): 108.



10.22214/IJRASET

45.98

IMPACT FACTOR:
7.129

IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)