**SYNOPSIS**

**PSEUDO SPYDER-APPLICATION FOR THE DETECTION OF FAKE NEWS**

Technology which is getting updated day-by-day has made the work load of human more easier and effort less. Also, social media networks which were created for the purpose of recreation and socialization over internet has become an inevitable part of our life. Now a days, rather than entertainment, more often social media is used for checking out news feeds.

But the trustworthiness of source and content of news that is being viewed and shared over the internet is a greater matter of concern. Because, pseudo news and hoaxes are spreading in a menacing manner like a pandemic which is yet to be sorted out. Without checking the reliability of source and without knowing whether the news is reel or real people used to share news which seems to be satisfactory, without realizing the fact that these incognizant acts may abate the power of fake news and may end up in a severe issue especially during the time of disaster.

“During 2017, as the united states prepared for hurricane Irma, an array of false information spread online. This included a Facebook post which falsely claimed the storm would hit Houston with a map showing of 14-day forecast-9 days longer than official forecasts,” the alert noted. “Within 24 hours, the national weather service publicly debunked the forecast on twitter, but the post had already been shared over 36,000 times on Facebook”.

These instances actually give a clear-cut idea about the danger hidden within fake news and the impact created by the same. So, it is high time to “think before click” i.e., to fact check the news and its reliability before sharing. For the above-mentioned purpose, we design an application to check whether a news is fake or real so that users can share reliable information which ensures protection from critical false news disasters. Our system identifies a news posted on the application as real or not.

We have a set of trusted sites such as Google. In these sites, if we search for a real content relevant information will be provided. If we search for a fake content, mismatching and irrelevant information will be retrieved as output. In this application, the news to be checked is searched online and results are gathered through web scrolling i.e., from different sites or from different links of Google. Then, the content obtained through search and the content of the concerned news is compared. comparing and finding out similarities by understanding contents is not an easy task. For that, we use techniques like natural language processing and associated set of algorithms. If similarities are found then that would be a real news and if not, it would be a fake news.

Thus, one can easily figure out fake news and break the chain of fake news from being shared over the internet globally

**CHAPTER I**

**INTRODUCTION1.1 PSEUDO SPYDER: AN APPLICATION FOR THE DETECTION OF FAKE NEWS - AN OVERVIEW**

As we know fake news are deliberate disinformation or hoaxes spread via traditional news media or social media. Fake news may be a relatively new term but it is not necessarily a new phenomenon. Fake news has technically been around at least since the appearance of partisan newspapers in the 19th century.

Advancement in technology and the spread of news through different types of media such as internet have increased the spread of fake news today.

As such, the effects of fake news have increased exponentially in the recent past and something must be done to prevent this from continuing in the future

Now a day’s fake news is spreading like a pandemic which is yet to be sorted out.

Only provision to distinguish between fake and real news are

1. Evaluating by the themselves

2. Checking online

3. Local news report

But often these methods fail to give accurate result

In the circumstance the need or relevance of an application solely fake news is revealed.

**1.2 BACKGROUND STUDY**

**1.2.1 THE ORGANIZATIONAL PROFILE**

REGIONAL TECHNOLOGIES is a rapidly growing company that provides professional IT services. they are one of the largest and best software development companies in Kerala with focus on .NET, .php, .java, software testing's and web design. As a leader in providing offshore software development and related services, REGIONAL TECHNOLOGIES function from an offshore setup based in Kerala, India.

Their reputation enables them to serve in terms off out sourced software development, web development, designing off web site and their corresponding development. Particularly high-end developments using Microsoft, NET, JAVA J2EE platform, ASP, ASP.NET, PHP development as well as VB.6.0 development adjoins us to a particularly niche sector off the global servicing.

They have worked for and have provided services to the 6 off the top 10 fortune 500 companies off the world they aim to provide their clients with comprehensive, end-to-end technology solutions that give them and advantage over the competition. From building applications that increase our productivity, to internet enabling our business for maximum profit.

Their mission is to continuously optimize their customers business through their world class solutions, services and products.

**1.2.2 STUDY ON EXISTING SYSTEM**

Social media networks which were created for the purpose of recreation and socialization over internet has become an inevitable part of our life. Now a days, rather than entertainment, more often social media is used for checking out news feeds.

But the trustworthiness of source and content of news that is being viewed and shared over the internet is a greater matter of concern.

Because, pseudo news and hoaxes are spreading in a menacing manner like a pandemic which is yet to be sorted out.

Without checking the reliability of source and without knowing whether the news is reel or real people used to share news which seems to be satisfactory. These blindly deeds make the one who shares the fake news as culprit without any acknowledgement and this incognizant act may upsurge the power of fake news and may end up in a severe issue especially during the time of disaster.

For example, in India, Misinformation related to coronavirus COVID-19 pandemic is in the form of social media messages related to home remedies that have not been verified, fake advisories and conspiracy theories. Also, fake news regarding economic crisis, monsoon floods and election is on its high range.

Therefore, it is high time to “think before click” That is, to fact check the news and its reliability before sharing. But differentiating fake and real news is quite a challenging task.

**1.2.3 DISADVANTAGE OF EXISTING SYSTEM**

1. Ambiguities on reliability and trustworthiness of content and its source.
2. Satisfactory appearance creates controversy.
3. Difficult to differentiate fake and real news at single sight.
4. Evaluating news to the best of our knowledge will be confusing and is subjective in nature.
5. Checking them online will be time consuming and may not be genuine always.
6. Fact-checking online will provide many results, Reading and analyzing them then interpreting into a conclusion is a tedious job.
7. Long procedures end up in lethargic mentality which provokes people to share what they got without bothering about its genuinely.
8. Results are not readily available.
9. User needs to refer multiple application and websites to check every single news.
10. No user-friendly, interactive, single-standard interface is available.
11. Local news channels and newspaper may indicate about trending fake news but it cannot be accessed 24x7 and also it will not accommodate all the news user wants.

**CHAPTER II**

**SYSTEM ANALYSIS**

**2.1 STUDIES ON PROPOSED SYSTEM**

Our application helps to check whether news is fake or real so that users can:

1. share reliable information
2. Ensures protection from critical false news disasters
3. Protect the users from being a culprit for sharing fake news.

We have a set of trusted sites such as Google. In these sites, If we search for a real content on them relevant information will be provided. If we search for a fake content, mismatching and irrelevant information will be retrieved as output.

Basic Working principle:

1. The news to be checked is searched online and results are gathered through web scrolling that is, from different sites or from different links of Google.
2. The content obtained through search and the content of the concerned news is compared.
3. Comparing and finding out similarities by understanding contents is not an easy work. For that, we use techniques like natural language processing and associated set of algorithms.
4. If similarities are found then that would be real news and if not, it would be a fake news.

We also implement “chat bot” to seek out reliable information to help users either confirm or deny what they’ve heard.

Enabling chatbot as a feature helps in:

1. Engaging large no. of people
2. Limiting human interventions
3. creating a more personalized , real-time and two-way experience
4. Creating easier , friendly and instant to use communication platform

**2.1.1 ADVANTAGES OF PROPOSED SYSTEM**

* Reliability and trustworthiness of content and its source is ensured.
* Non-technical i.e., appearance based valuation of news is completely omitted.
* One click is enough to differentiate between fake and real news.
* Scientific comparison and analysis is done by avoiding legacy subjective examination.
* Instant, Real-time and genuine result is obtained.
* Simple and easy to use since, long procedures are excluded.
* All the required features and integrated into a single platform so that no need to refer multiple websites and applications
* Standard, easy to use and user friendly interface is implemented with the help of chat bots.
* Inorder to clarify doubts about the application full time assistance is provided by chatbot.
* No time constraints to fact check news since, it is available 24x7.
* Accurate results are obtained with less time consumption and least usage of resources.
* News can be fact-checked according to user’s demand.
* User can share news among friend circle easily since, they are provided with features such as sending and accepting friend request and chats.

**2.2 USER REQUIREMENT SPECIFICATION**

There are two modules:

1. Admin
2. User

**Function of modules**

**1. Admin**

* + Login
  + View users
  + Block users
  + Managing feedbacks
  + Adds questions and answers to chatbot
  + Overall Chatbot management

**2. User**

* Register
* Login
* Fake news detection
* Views and posts news
* Sends and accepts friend request
* Chat with friends
* Provides feedback
* Communicate with chat bot

**2.3 SOFTWARE REQUIREMENT SPECIFICATION**

**2.3.1 PYTHON**

Python is interpreted, high-level and open-source programming language. It is one of the top rated programming language now a days. It holds a vital place among the top tools for Data Science and is often the go-to choices for advanced IT concepts such as Artificial intelligence, Machine learning ,deep learning and much more.

It is a general purpose programming language which supports various concepts of object oriented programming and is portable and interactive in nature. Rich set of libraries, User friendly data structure, dynamically typed language are some of the features of python. Of course, python is user friendly and easy to learn language.

**Why python in this project?**

First of all we consider this project as an opportunity to learn and explore programming language especially trending one such as python, as well as its different dimensions. Python is general purpose programming language so that, it was easy to integrate webpage designs, background coding for website and server side coding for android under one roof. Since python is associated with rich set of frameworks such as flask (for web development), inbuilt packages such as JavaScript object notation (json, standard format for representing structured data) and interfaces such as pymysql ( which makes smooth and easy connection with database) it provided utmost comfort and understandability for working. Python also sub structured basic machine learning concept effectively and efficiently with the help of packages such as NLTK (Natural Language Tool Kit).

**2.3.2 MySQL**

MySQL is a database management system that allows us to manage relational databases. It is open source software backed by oracle. MySQL is pretty easy to master in comparison with other database software like Oracle database, or Microsoft SQL Server. MySQL can run on various platforms UNIX, Linux, Windows etc. You can install it on a server or even in a desktop. Besides MySQL is reliable, Scalable and fast.

Since, we are much used to MySQL than any other RDBMS software it was pretty much easier to work with and we got a chance to explore more on SQL queries and dependencies especially on complicated ones such as join query.

**2.3.3 XAMPP**

XAMPP is an abbreviation for cross-platform, Apache, MySQL, PHP and Perl. Itis a software distribution which provides the Apache web server, MySQL database (actually MariaDB), Php and Perl (as command-line executables and Apache modules) all in one package. It is available for Windows, MAC and Linux systems. No configuration is necessary to integrate Php with MySQL.

It is a great fit for this project and provides a relatively painless installation and way to manage the configuration changes. Also provided is PhpMyadmin which gives a GUI tool for managing your MySQL databases.

**2.3.4 SQLyog**

SQLyog is the most powerful manager, admin and GUI tool for MySQL, combining the features of MySQL Query Browser, Administrator, phpMyAdmin and other MySQL Front Ends and MySQL GUI tools in a single intuitive interface. SQLyog is a fast, easy to use and compact graphical tool for managing your MySQL databases. SQLyog was developed for all who use MySQL as their preferred RDBMS. Whether you enjoy the control of handwritten SQL or prefer to work in a visual environment, SQLyog makes it easy for you to get started and provides you with tools to enhance your MySQL experience.

SQLyog was included as a part of our project because, we were able to learn new queries and dependencies and SQLyog integrated all the tools required for the effective management of MySQL RDBMS under one shelter. All the tables involved in our project and their structure was organized in a representable and understandable manner with the help of user friendly interfaces. Moreover, we were able to create tables and manage those using editable queries in an easier way when compared to that of handwritten SQL commands.

**2.3.5 ANDROID**

Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, developed by Google. It was designed primarily for touchscreen mobile devices such as smartphones and tablets. Unlike Apple’s ios android is open source and Developers can create programs for Android using the free Android software developer kit ([SDK](https://techterms.com/definition/sdk)). Android programs are written in [Java](https://techterms.com/definition/java) and run through a Java virtual machine [JVM](https://techterms.com/definition/jvm) that is optimized for mobile devices called DVM (Dalvik Virtual Machine). The current stable version is [Android 11](https://en.wikipedia.org/wiki/Android_11), released on September 8, 2020.

**Why android in this project?**

Unlike Apple’s ios Android is more flexible and functional, most importantly android is open source when compared to ios. In this era of forward moving technology apps and app development play a vital role even in everyday life. We opt an application in the user’s phase because, apps are more easy and simple to handle than website even if the user doesn’t have that much technical knowledge. The developing environment that is, the IDE used was ANDROID STUDIO which was easy to handle, understand and work with. This project helped us to learn various aspects and concepts of application development and also, Developing attractive user interface and running our project on a real device and on real time was definitely a matter of excitement.

* + 1. **WINDOWS 7**

Windows 7 is a version of Windows that succeeded [Windows Vista](https://en.softonic.com/downloads/windows-vista). Compared to its predecessor it provides **high speeds and streamlines the interface**while maintaining valuable features such as the User Account Control [sandbox](https://en.softonic.com/downloads/sandbox) and the aero graphical enhancements. In many respects, it is very moderate sitting at the center of a web of operating systems with many different traits. Windows 7 operating system has advantages such as High software compatibility, High speed, Secure and supported minimum interruptions.

* 1. **FEASIBILITY STUDY**

A feasibility study is an analysis that takes all of a project's relevant factors into account including economic, technical, legal, and scheduling considerations, to ascertain the likelihood of completing the project successfully. Project managers use feasibility studies to discern the pros and cons of undertaking a project before they invest a lot of time and money into it. Feasibility studies also can provide a company's management with crucial information that could prevent the company from entering blindly into risky businesses.Some of the key factors of feasibility study such as, problem or opportunity to be studied, an analysis of the current mode of operation, a definition of requirements, and an evaluation of alternatives have been done in our project.

**2.4.1 ECONOMIC FEASIBILITY**

In Economic Feasibility study, cost and benefit of the project is analyzed. Means under this feasibility study a detail analysis is carried out what will be cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on. After that it is analyzed whether project will be beneficial in terms of finance for organization or not.

Our project actually uses minimal set of hardware and software resources in an effective manner. Also, all the software tools that assisted the development of the application were chosen open source. So that cost for paid versions of those tools can be omitted. Even though open source tools were used, quality and efficiency is not at all compromised thus, we can surely say our project is cost effective. Also different phases of the project are completed with least usage of resources without any additional overhead. Thus, we can surely say our project is cost effective and has performance worth for the cost invested.

**2.4.2 TECHNICAL FEASIBILITY**

In Technical Feasibility current resources both hardware software along with required technology are analyzed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyzes technical skills and capabilities of technical team, existing technology can be used or not, maintenance and up-gradation is easy or not for chosen technology etc.

Technical skills regarding the project can be easily acquired because, Language such as python are easy and interesting to learn. Minimal hardware requirements such as computer with sufficient, least number of peripherals are only necessary when it comes to hardware requirements. But while running slightly heavier development programs like android studio it is preferred to have 8GB RAM for smooth working. Even though it seem to be like an additional overhead it is neutralized by the smooth performance and efficiency ensured by the same. Easily and readily available software tools and os such as Windows 7 which satisfies basic need is adequate. Thus technical feasibility of our project is ensured.

**2.4.3 OPERATIONAL FEASIBILITY**

In Operational Feasibility degree of providing service to requirements is analyzed along with how much easy product will be to operate and maintenance after deployment. Along with this other operational scopes are determining usability of product, Determining suggested solution by software development team is acceptable or not etc.

One of the key goals of our project is to simplify the complexity of the task thus, our project ensures usability even for people who has no deep technical knowledge. Since, all the functionalities are integrated within single platform maintainability is easier. Modifications or updation can be easily accommodated that is, new feature or modules can also be added in a simpler manner.

* 1. **SYSTEM SPECIFICATION**

**2.5.1 HARDWARE SPECIFICATION**

* PROCESSOR : Intel dual core or above
* HARD DISK : 160 GB
* RAM : 4 GB
* INPUT DEVICES : Keyboard , Mouse
* OUTPUT DEVICES : Monitor

**2.5.2 SOFTWARE SPECIFICATION**

* OPERATING SYSTEM : Windows 7 or above
* TECHNOLOGY : Python , Android
* BACK END : MySQL
* PLATFORM USED : JetBrains PyCharm, Android studio
* WEB BROWSER : Google chrome

**2.6 COST ESTIMATION AND SCHEDULING**

**2.6.1 COST ESTIMATION**

* Man power = Rs.3000.00
* Electricity charge per day =Rs.10.00
* Internet cost per day =Rs.07.00
* Total page cost =Rs.1500.00
* Overhead =Rs.2000.00

**TOTAL =Rs.6517.00**

**2.6.2 SCHEDULE ESTIMATION**

|  |  |
| --- | --- |
| **PHASE** | **SCHEDULE** |
| Requirement Gathering | 9 Days |
| Requirement Analysis | 10 Days |
| Design | 6 Days |
| Coding | 20 Days |
| Testing | 10 Days |
| Deployment | 10 Days |
| **TOTAL** | **65 Days** |

**CHAPTER III**

**DESIGN AND DEVELOPMENT PROCESS**

**3.1 FUNDAMENTAL DESIGN CONCEPTS**

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.If the broader topic product development "blends the perspective of marketing, design, and manufacturing into a single approach to product development," then design is the act of taking the marketing information and creating the design of the product to be manufactured. Systems design is therefore the process of defining and developing systems to satisfy specified requirements of the user. The design concepts provide the software designer with a foundation from which more sophisticated methods can be applied. A set of fundamental design concepts has evolved. They are as follows:

1. **Abstraction**- Abstraction is the process or result of generalization by reducing the information content of a concept or an observable phenomenon, typically in order to retain only information which is relevant for a particular purpose.
2. **Refinement**- It is the process of elaboration. A hierarchy is developed by decomposing a macroscopic statement of function in a step-wise fashion until programming language statements are reached. In each step, one or several instructions of a given program are decomposed into more detailed instructions. Abstraction and Refinement are complementary concepts.
3. **Modularity**- Software architecture is divided into components called modules.
4. **Software Architecture**- It refers to the overall structure of the software and the ways in which that structure provides conceptual integrity for a system. Good software architecture will yield a good return on investment with respect to the desired outcome of the project, e.g. in terms of performance, quality, schedule and cost.
5. **Control Hierarchy** - A program structure that represents the organization of a program component and implies a hierarchy of control.
6. **Structural Partitioning** - The program structure can be divided both horizontally and vertically. Horizontal partitions define separate branches of modular hierarchy for each major program function. Vertical partitioning suggests that control and work should be distributed top down in the program structure.
7. **Data Structure** - It is a representation of the logical relationship among individual elements of data.
8. **Software Procedure** - It focuses on the processing of each module individually.
9. **Information Hiding** - Modules should be specified and designed so that information contained within a module is inaccessible to other modules that have no need for such information.

**3.2 DESIGN NOTATION**

**3.2.1 UML DIAGRAM (USE CASE DIAGRAM)**

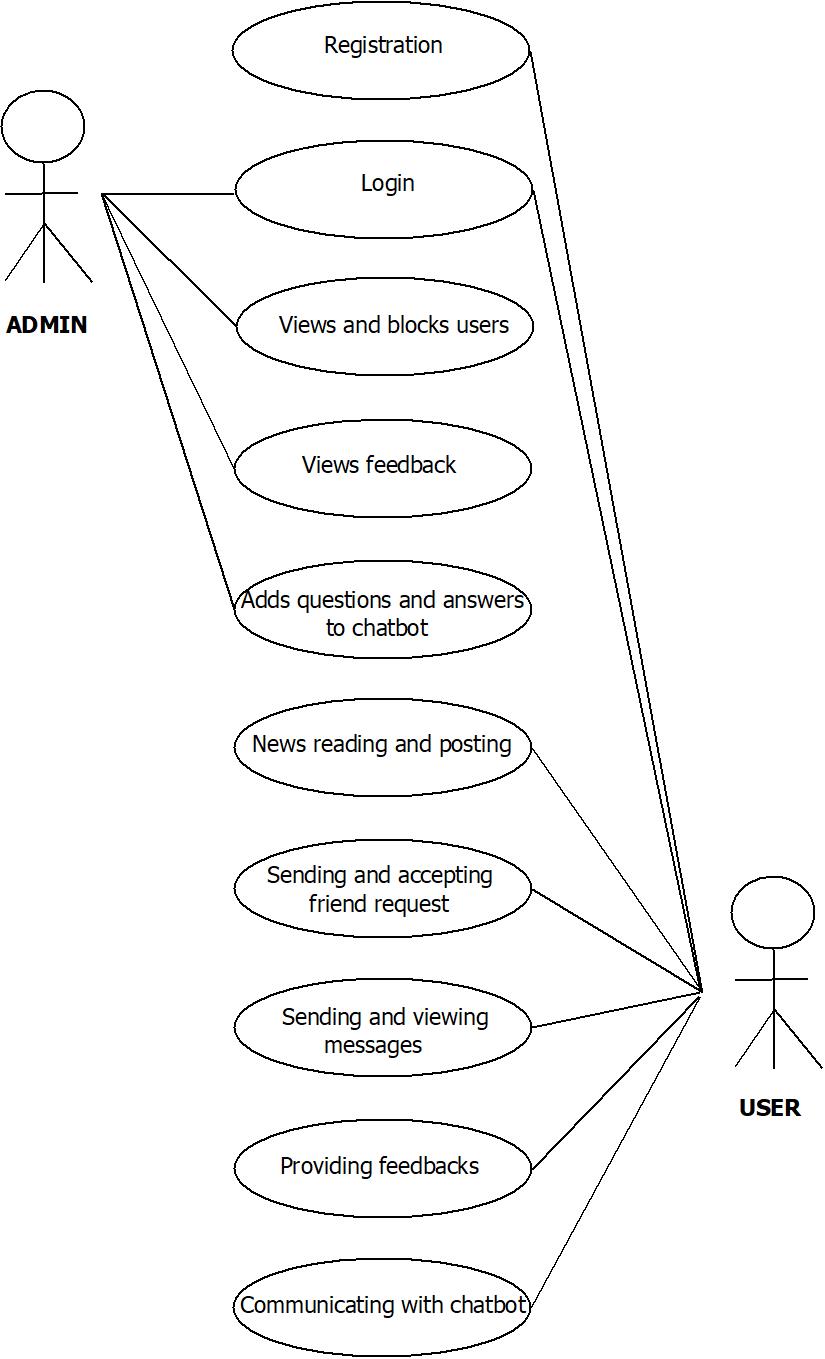
A UMLdiagram is a diagram based on the UML (Unified Modelling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.

A use case diagram is a representation of user’s interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. It models how external entity interacts with the system to make it work

Notations used in use case diagram are:

ACTOR

USECASE/FUNCTIONALITY



**3.2.2 DATA FLOW DIAGRAM**

DFD is a structured analysis modelling tool where the whole system is subdivided and then analysed. The flow of data of a system or a process is represented and it helps to visualize major steps involved in software or system process. DFD has no control flow, loops or decision making as in flowchart. All details about data such as where it comes, where it goes, how it is stored is represented by its four major elements they are Entity/external entity, Dataflow, Data store/Ware house and process.

Notations used in DFD are:

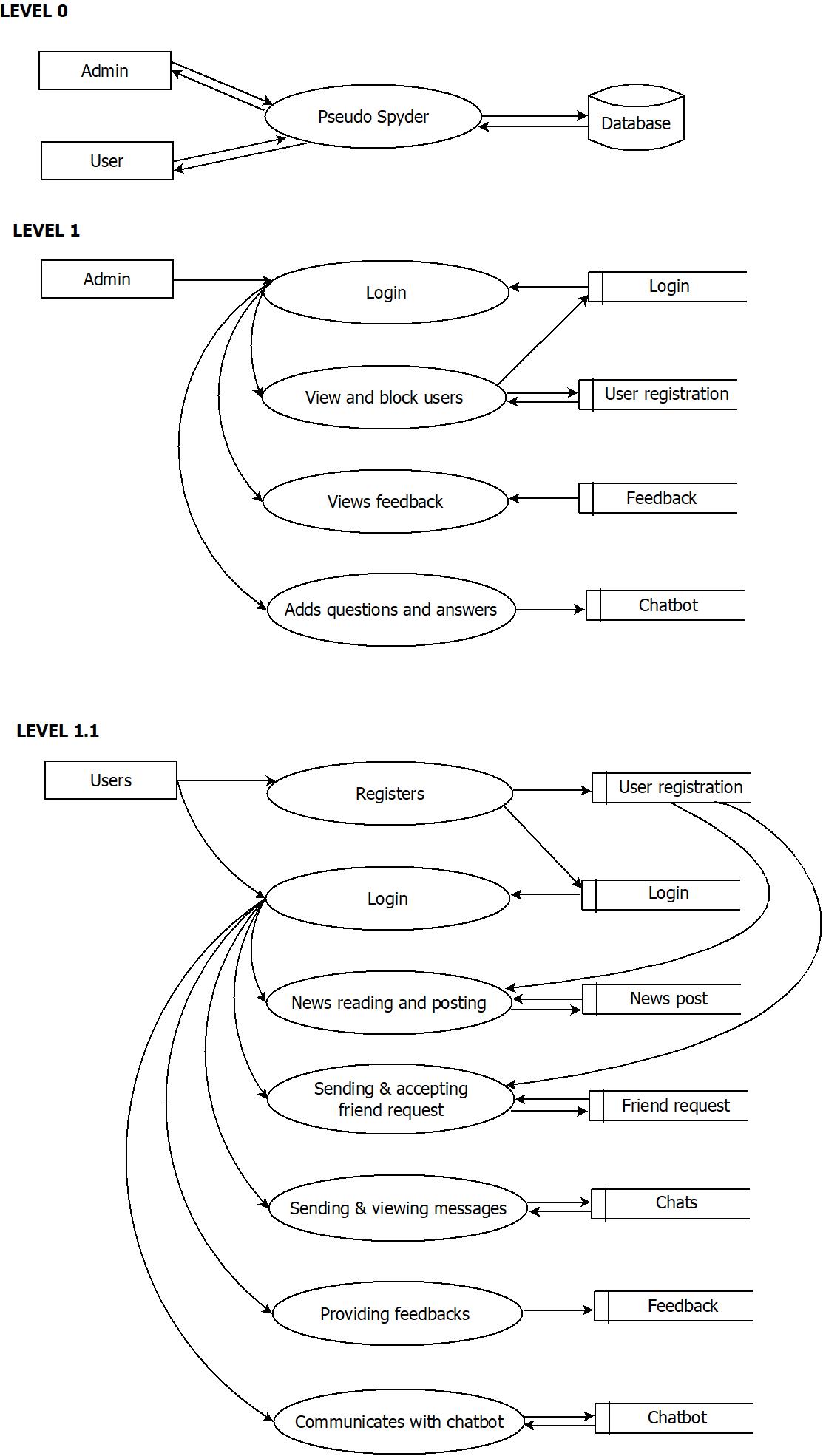
PROCESS

DATAFLOW/FLOW LINE

EXTERNAL ENTITY

DATASTORE

* Process : Transforms incoming dataflow to outgoing dataflow
* Dataflow : Pipeline through which data flows
* External entity : Objects outside the system and those interact with the system
* Data store : Repository of data in the system



**3.2.3ER DIAGRAM**

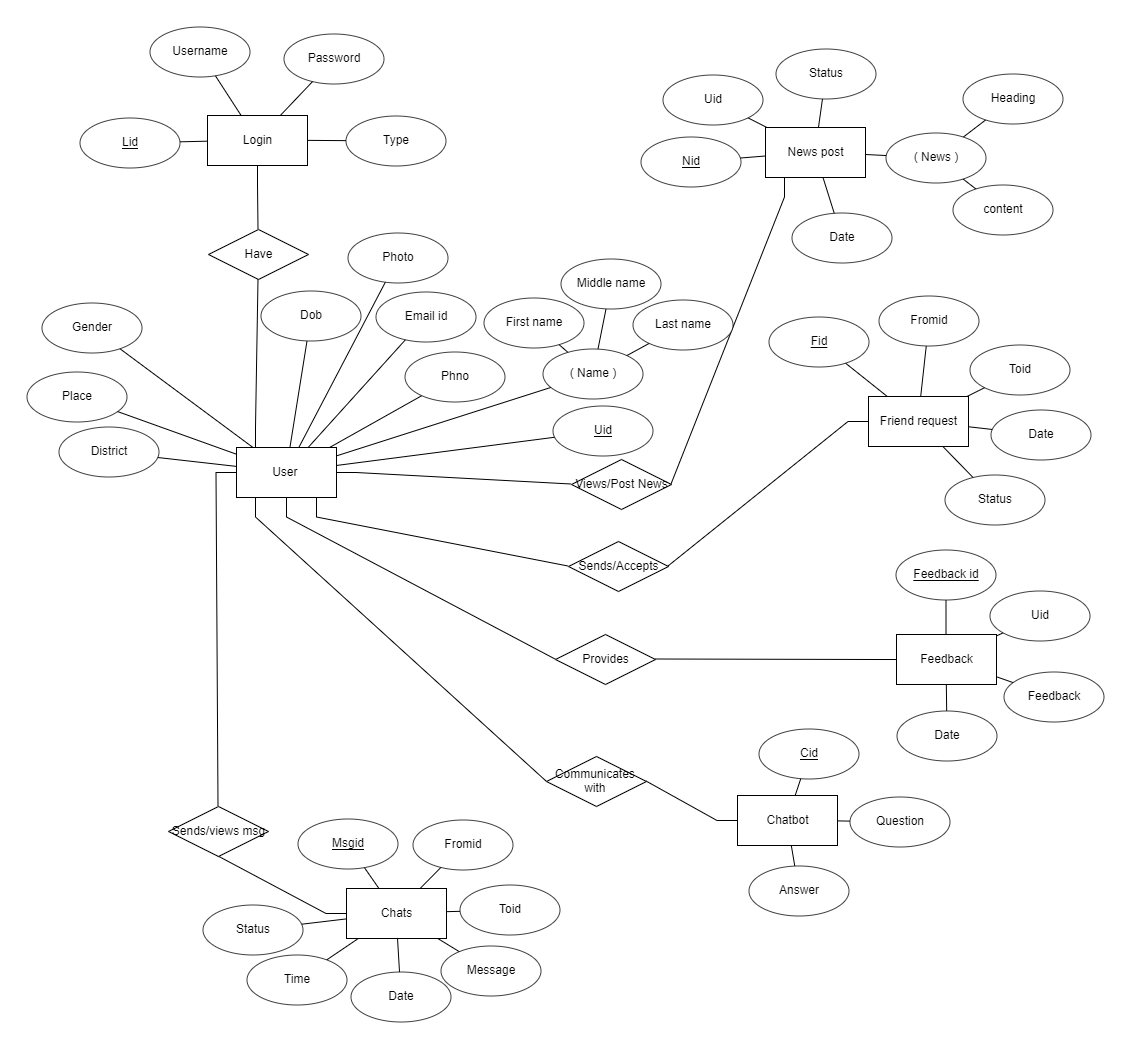
**ER Diagram** stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases.

Terminologies of ER diagram are:

* Entity – Object in real world
* Attributes – Properties of entity
* Entity set – Collection of entities with similar attributes
* Relation – Association between entity set

Symbols used in ER diagrams are:

|  |  |
| --- | --- |
|  | ENTITY SET |
|  | RELATIONSHIP |
|  | ATTRIBUTE |
|  | FLOW LINE |



**3.3DESIGN PROCESS**

**3.3.1 DATABASE DESIGN**

**1)Login table**

|  |  |  |
| --- | --- | --- |
| **FIELD** | **TYPE(SIZE)** | **SPECIFICATIONS** |
| Lid | int | Primary key  Auto increment |
| username | Varchar(20) | Not Null  Unique |
| password | Varchar(20) | Not Null  Unique |
| type | Varchar(10) | Not Null |

**2)User registration table**

|  |  |  |
| --- | --- | --- |
| **FIELD** | **TYPE(SIZE)** | **SPECIFICATIONS** |
| Uid | int | Primary key  Auto increment |
| First name | Varchar(20) | Not Null |
| Middle name | Varchar(20) |  |
| Last name | Varchar(20) |  |
| Phno | Bigint(20) | Not Null  Unique |
| Email id | Varchar(30) | Not Null  Unique |
| photo | Varchar(50) | Path of the image will be specified |
| Gender | Varchar(15) |  |
| Dob | Date |  |
| Place | Varchar(30) |  |
| District | Varchar(30) |  |

**3)Newspost table**

|  |  |  |
| --- | --- | --- |
| **FIELD** | **TYPE(SIZE)** | **SPECIFICATIONS** |
| Nid | Int | Primary key  Auto increment |
| Uid | Int | Foreign key |
| Heading | Text |  |
| Content | Text |  |
| Date | Date |  |
| Status | Varchar(10) |  |

**4)Friend Request table**

|  |  |  |
| --- | --- | --- |
| **FIELD** | **TYPE(SIZE)** | **SPECIFICATIONS** |
| Fid | Int | Primary key  Auto increment |
| Fromid | Int |  |
| Toid | Int |  |
| Date | Date |  |
| Status | Varchar(10) |  |

**5)Chats table**

|  |  |  |
| --- | --- | --- |
| **FIELD** | **TYPE(SIZE)** | **SPECIFICATIONS** |
| Msgid | Int | Primary key  Auto increment |
| Fromid | Int |  |
| Toid | Int |  |
| Message | Text |  |
| Date | Date |  |
| time | Time |  |
| Status | Varchar(10) |  |

**6)Feedback table**

|  |  |  |
| --- | --- | --- |
| **FIELD** | **TYPE(SIZE)** | **SPECIFICATIONS** |
| Feedback id | Int | Primary key  Auto increment |
| Uid | Int | Foreign key |
| Feedback | Text |  |
| Date | date |  |

**7)Dataset table**

|  |  |  |
| --- | --- | --- |
| **FIELD** | **TYPE(SIZE)** | **SPECIFICATION** |
| Cid | Int | Primary key  Auto increment |
| Question | Text |  |
| Answers | Text |  |

**8)Chatbot table**

|  |  |  |
| --- | --- | --- |
| **FIELD** | **TYPE(SIZE)** | **SPECIFICATION** |
| Id | Int | Primary key  Auto increment |
| U\_id | Int |  |
| Msg | Text |  |
| Answer | Text |  |

**3.3.2NORMALIZATION**

Normalization is the process of dividing or decomposing a table into different tables inorder to minimise redundancy and null values. Redundancy in relation may cause insertion, deletion and updation anomalies. So, it helps to minimize the redundancy in relations. Normal forms are used to eliminate or reduce redundancy in database tables. Different normal forms and conditions to attain the same are mentioned below:

1. First Normal Form

* The values of an attribute or column should be atomic that is indivisible or single valued.

2. Second Normal Form

* It has to be in first normal form
* All non-prime attributes are fully functional dependent on primary key or no partial dependency exist.
* All attribute those are part of primary key is called prime attribute others are called non-prime attribute.

3. Third Normal Form

* It should be in second normal form
* No transitive dependency should exist
* Transitive dependency defined as

A🡪B

B🡪C

A🡪C

This transitive dependency should not exist

4. Boyce-Codd Normal Form

* It should be in third normal form
* Defined over a set of relation R
* A🡪B, A,B⊆R
* A🡪B Then A is a super key or trivial dependency exist that is, B⊆A
* The above conditions are to be satisfied inorder to be in BCNF.

5. Fourth Normal Form

* Table should be in Boyce-Codd normal form
* No multivalued dependency should exist
* Multivalued dependency is defined such that, If A has multiple values on B and A has multiple values on C then B and C are related. This multivalued dependency should not be satisfied to attain fourth normal form

6. Fifth Normal Form

* Table has to satisfy the condition of fourth normal form
* Join dependency should not exist or Join should be lossless.

Our project is in Third normal form (3NF) that is, all the tables in our database design satisfy the condition of first, second and third normal form. The values of all attributes in all the tables are atomic so tables are in first normal form. Since, all the non-prime attribute in all the table are fully functional dependent on their concerned primary key then it is sure that all tables satisfies 2NF .Finally, Due to the absence of transitive dependency all the tables are in Third normal form. Since, some of the basic normal forms are achieved by all the tables in the database it is pretty much sure that redundant or null values are reduced effectively.

**CHAPTER V**

**CONCLUSION**

**5.1 FUTURE ENHANCEMENTS**

Any systems that have been in use for number of years are more prone to changes and amendments especially due to the fast evolution of science and technology and also due to the changing requirements. Likewise our project will be prone to changes in future, In such circumstances our project will be easily adaptable and modifiable. Also, one of the most exciting parts of our project is that it possesses a vast scope in future.

One such enhancement that can be brought to our project is that we can provide an advanced feature to examine news which is not published or readily available in any other primary sources. In such case, users will be provided with an option to post new news in the application which may include indigenous and isolated news which are difficult to reach the internet or even the outer world quickly. Here we access the messages or news from the user and verify with its location on the basis of a polling system.

Similarly, more advanced options can be included to our project that facilitates more effective barrier against fake news because, the era of fake news not going to end soon and there are possibilities that it may evolve in a more contiguous and deadly manner.

**5.2 BIBLIOGRAPHY**

**TEXTBOOKS**

* Software Engineering : A practitioner’s approach – 6th Edition : Roger S Pressman
* Android Programming – 2nd Edition : Joel Murach
* The Complete Reference Python – Martin C Brown

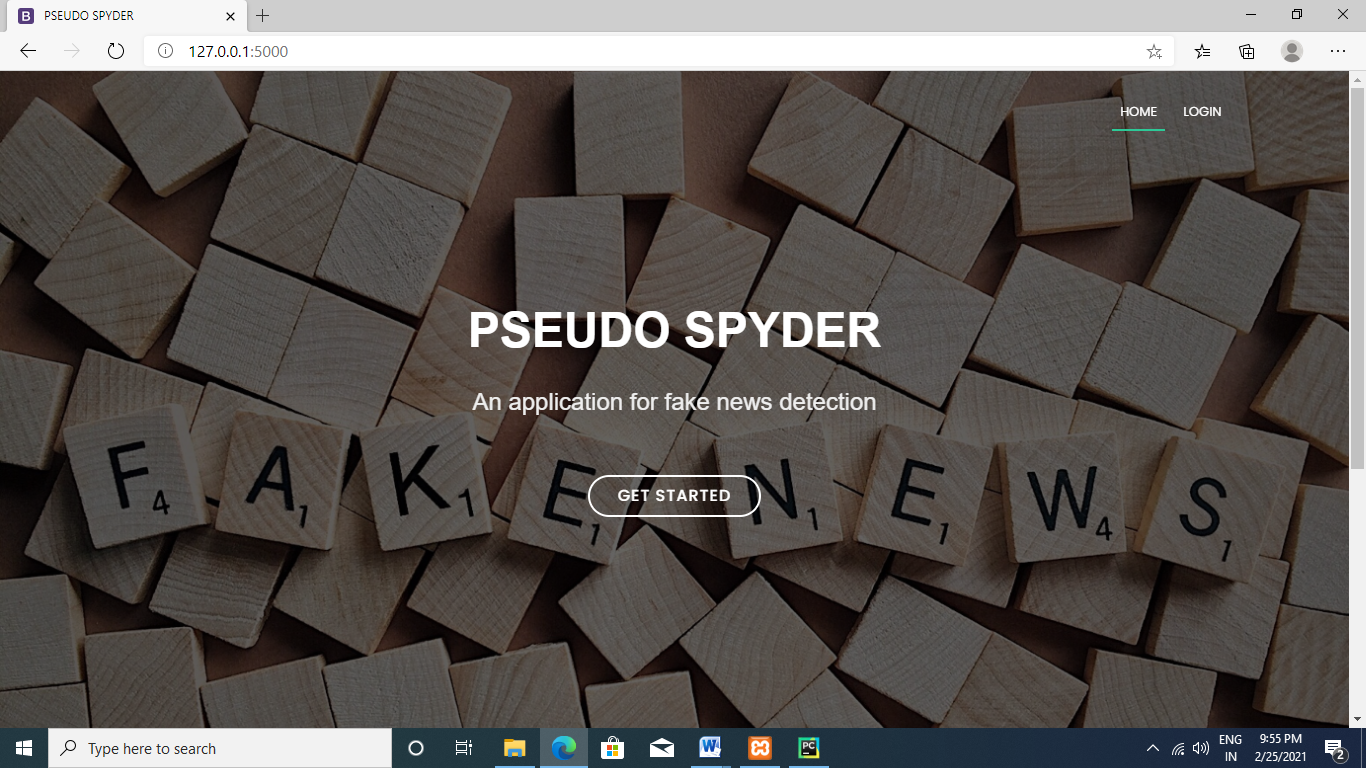
**WEB SITES**

* [www.w3schools.com](http://www.w3schools.com)
* [www.tutorialspoint.com](http://www.tutorialspoint.com)
* [www.geeksforgeeks.org](http://www.geeksforgeeks.org)
* [www.free-css.com](http://www.free-css.com)
* [www.python.org](http://www.python.org)

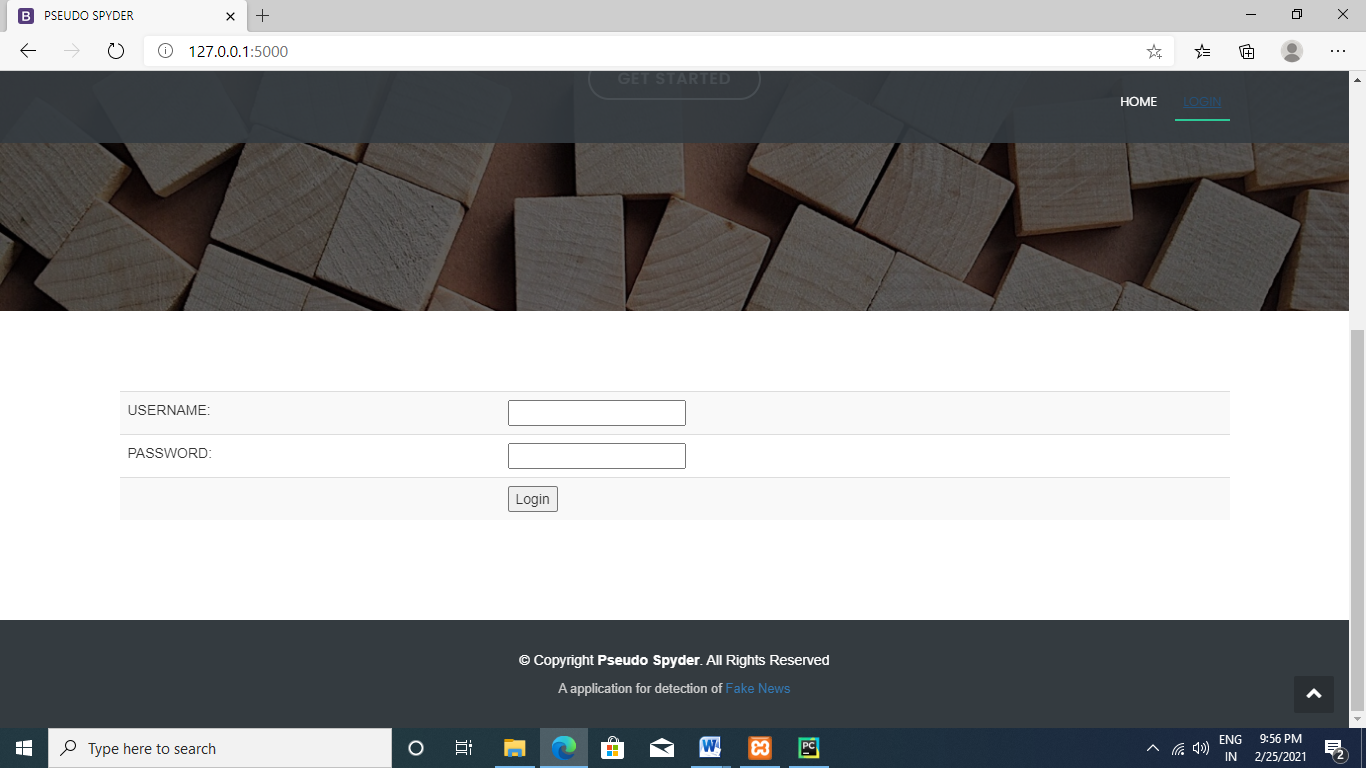
**5.3 ANNEXURE**

**5.3.1 WEB PART / ADMIN MODULE**

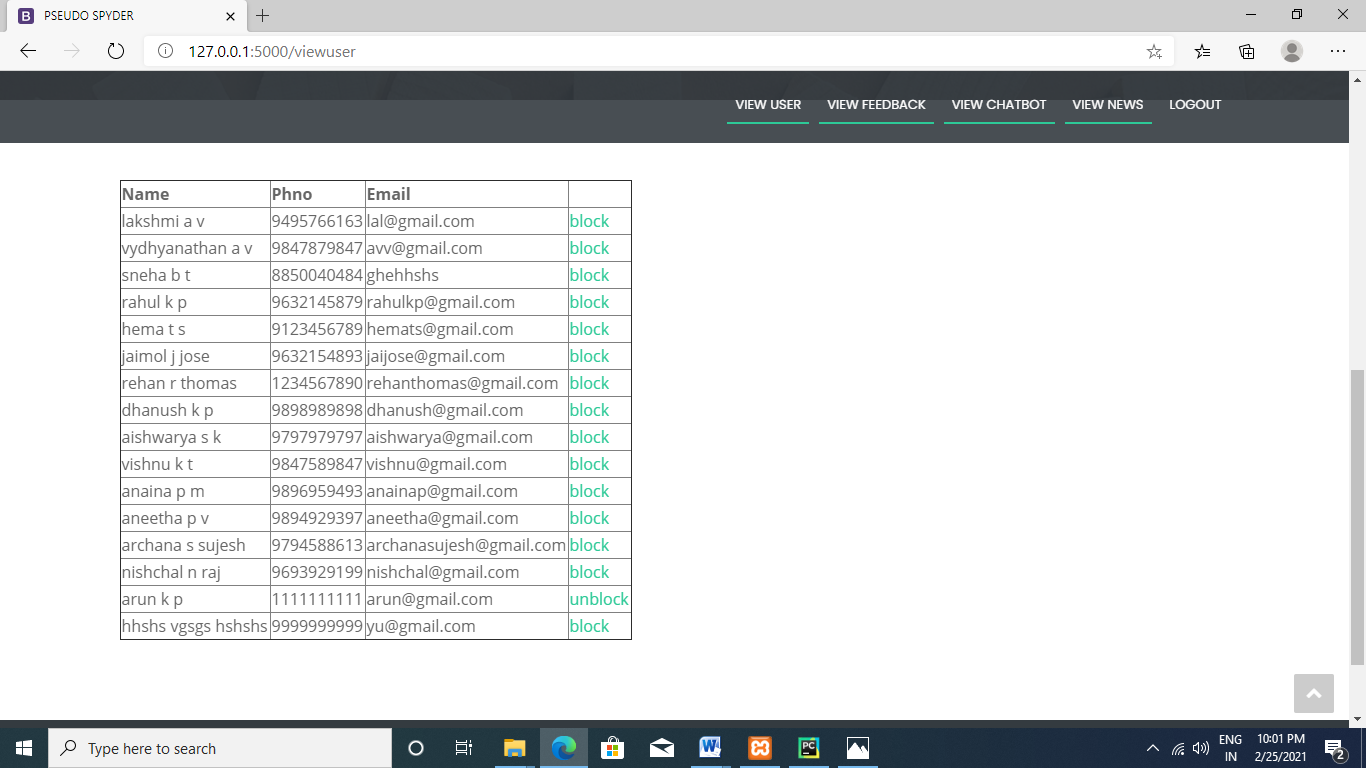
HOME PAGE



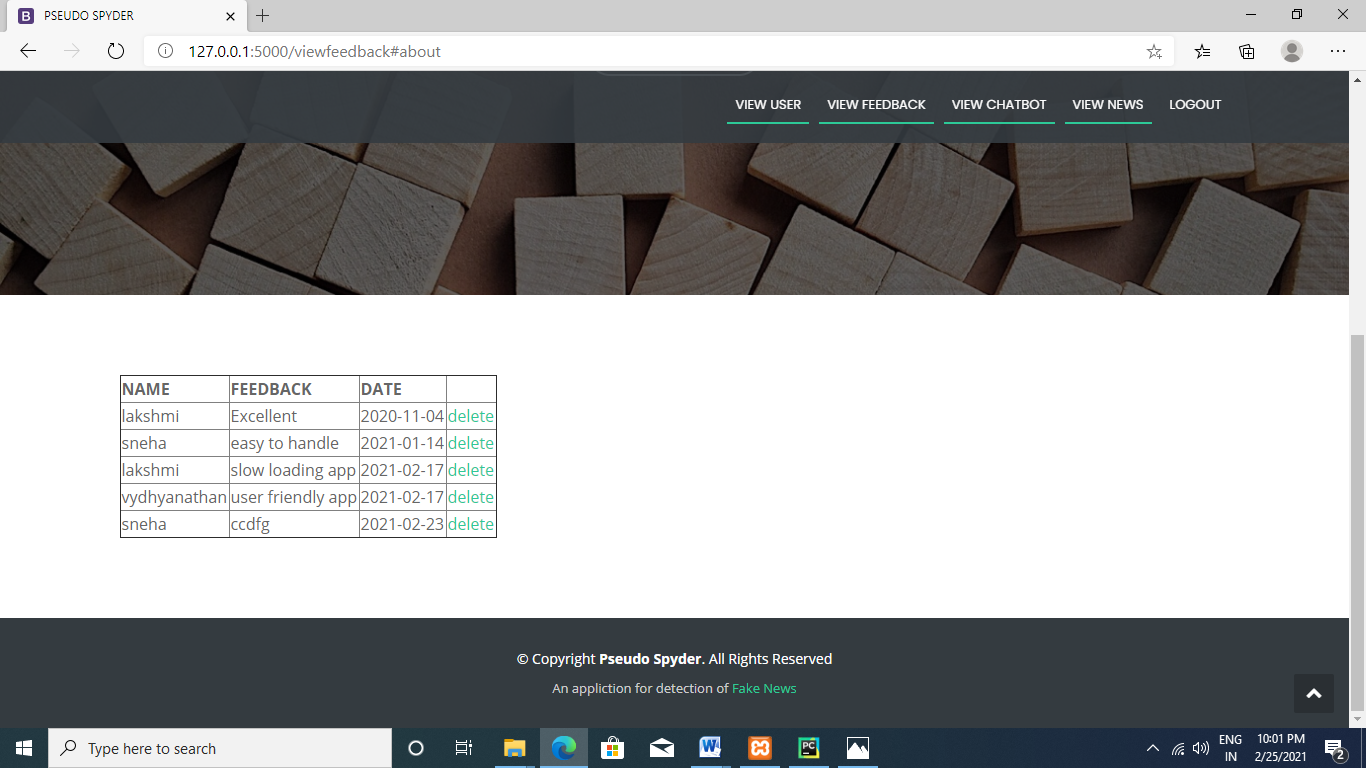
LOGIN PAGE

****

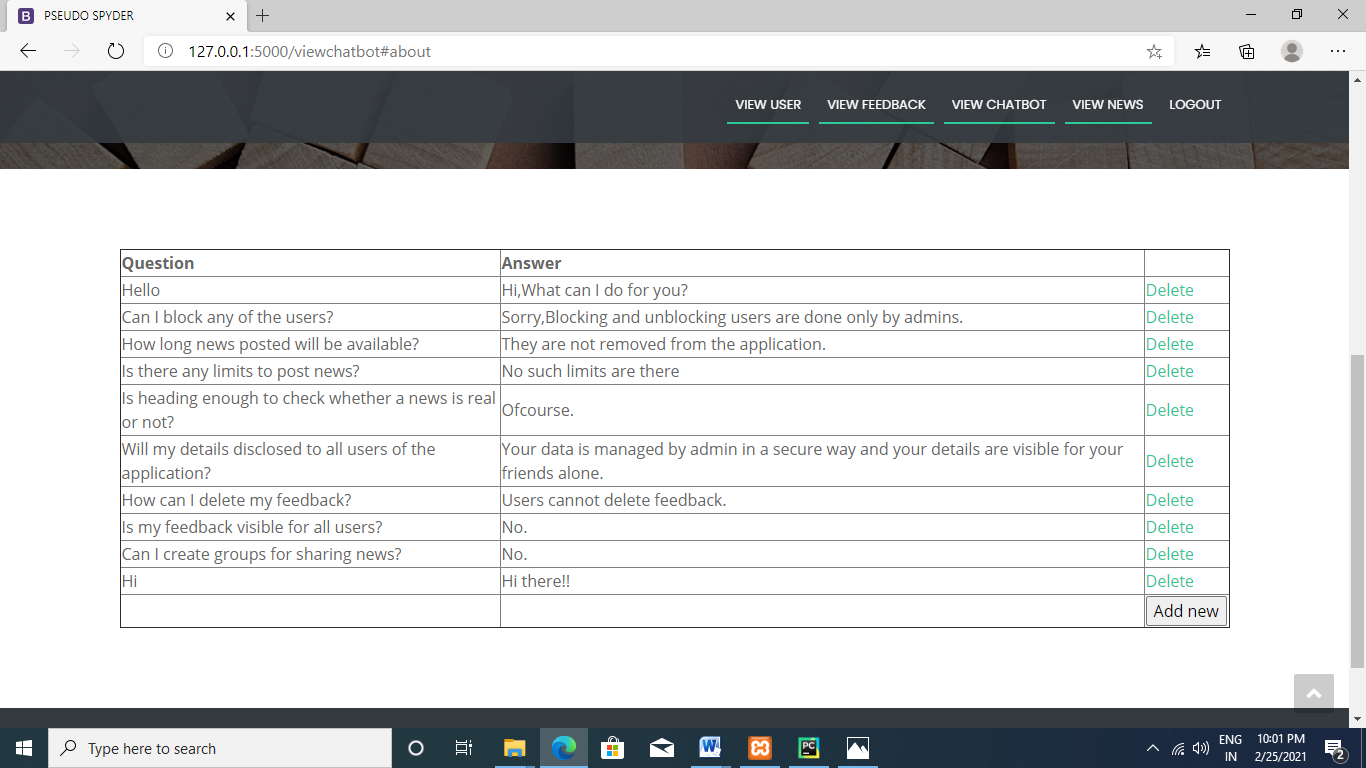
VIEW USER



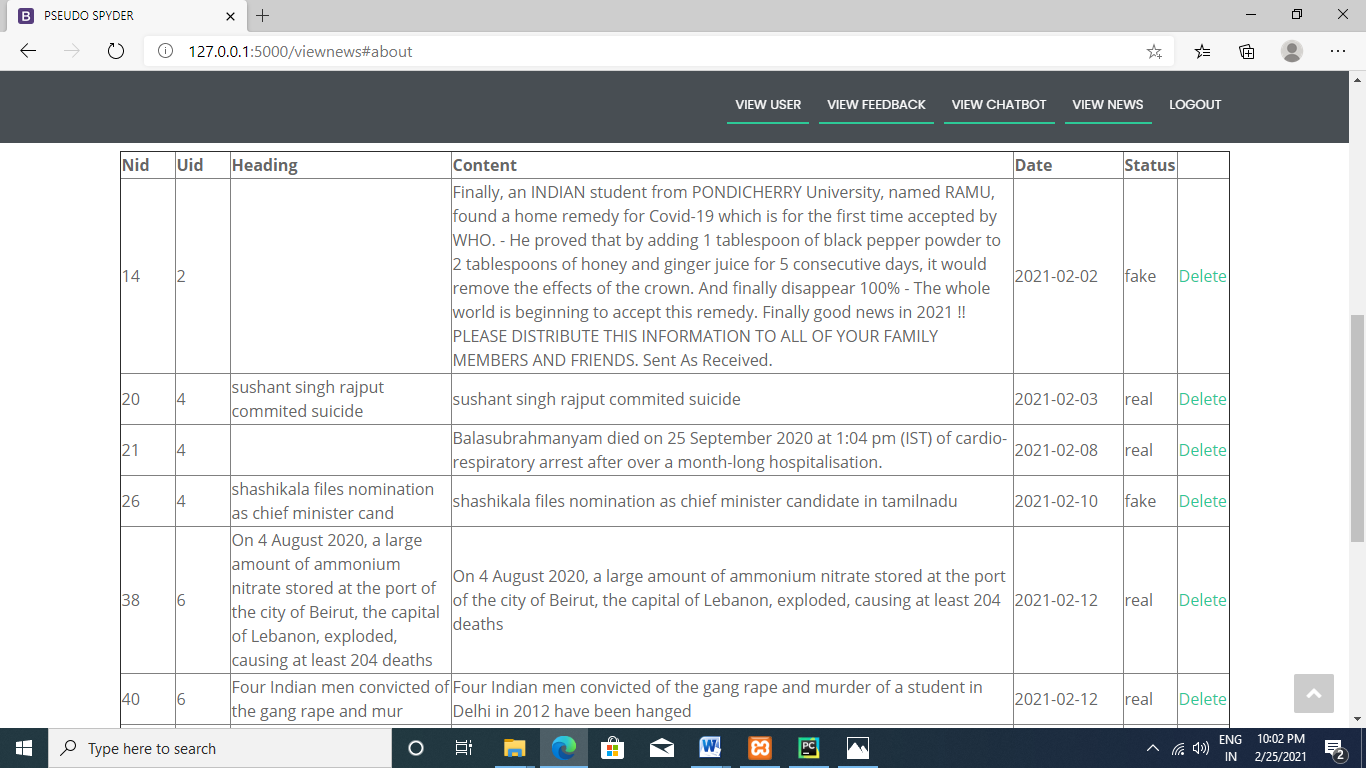
VIEW FEEDBACK



VIEW CHATBOT

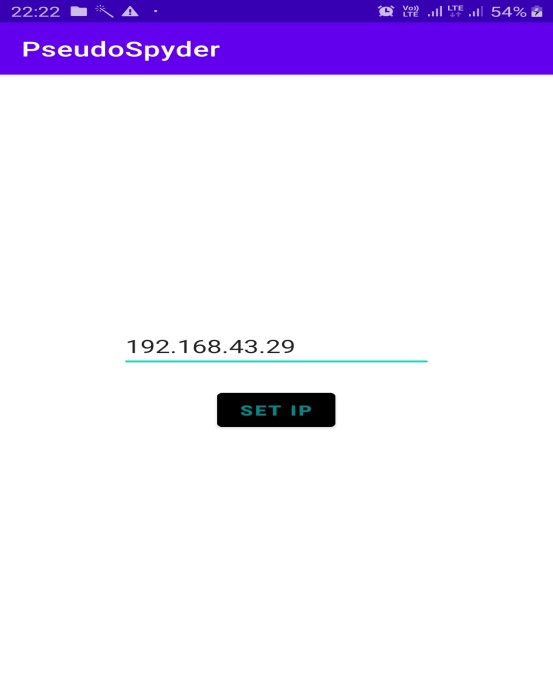


VIEW NEWS



**5.3.2 ANDROID PART / USER MODULE**

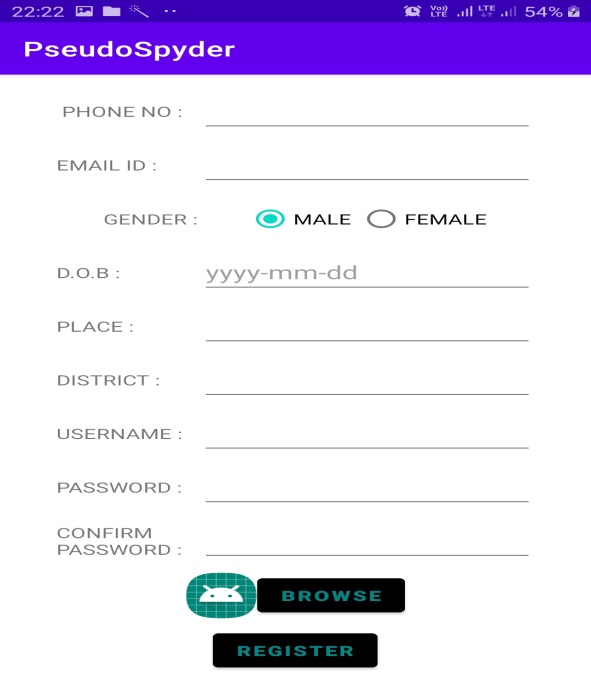
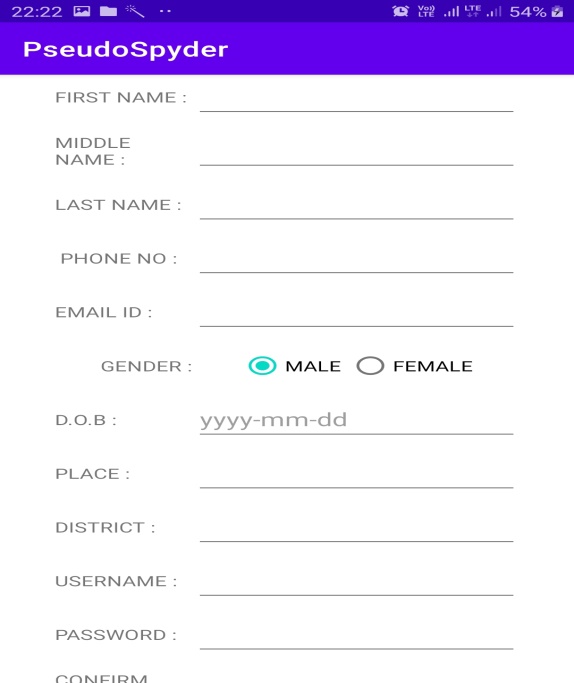
PAGE TO SET IP



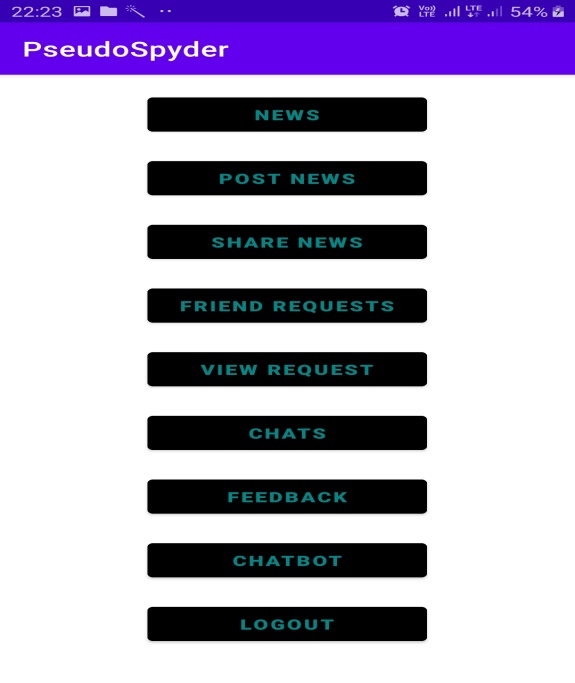
LOGIN PAGE



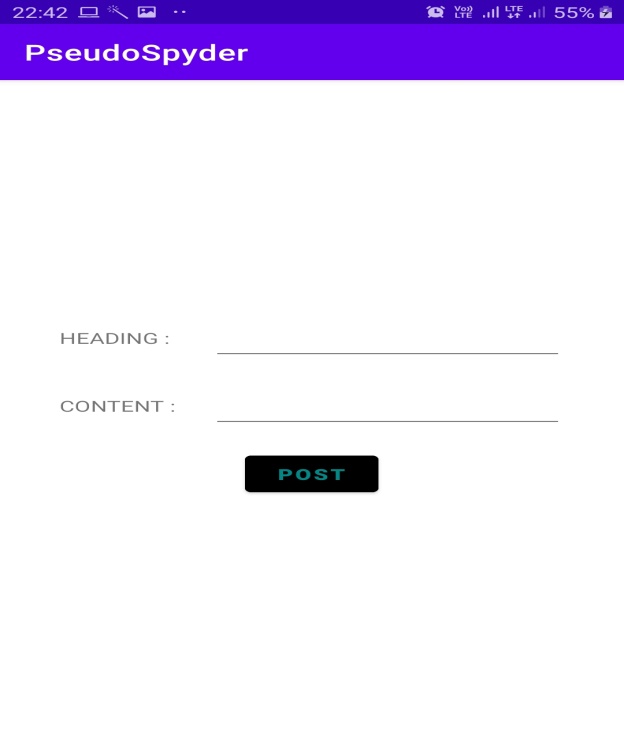
REGISTRATION PAGE



HOME PAGE



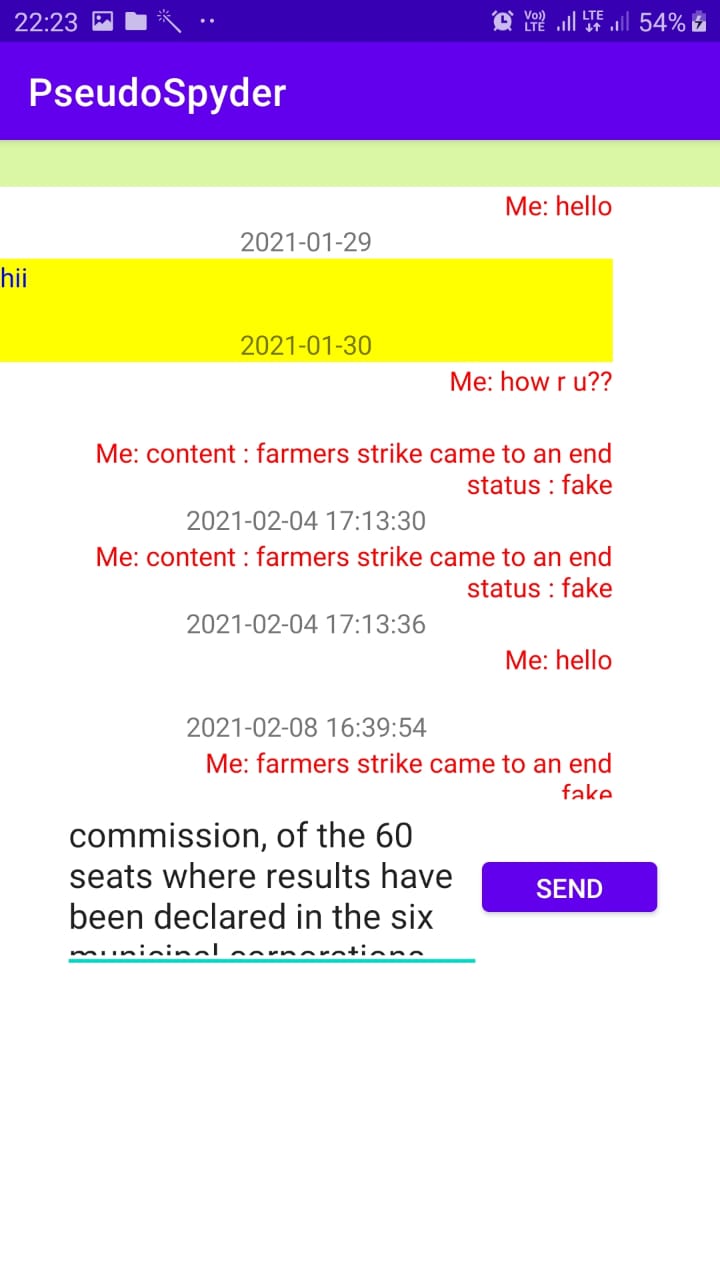
PAGE TO POST NEWS



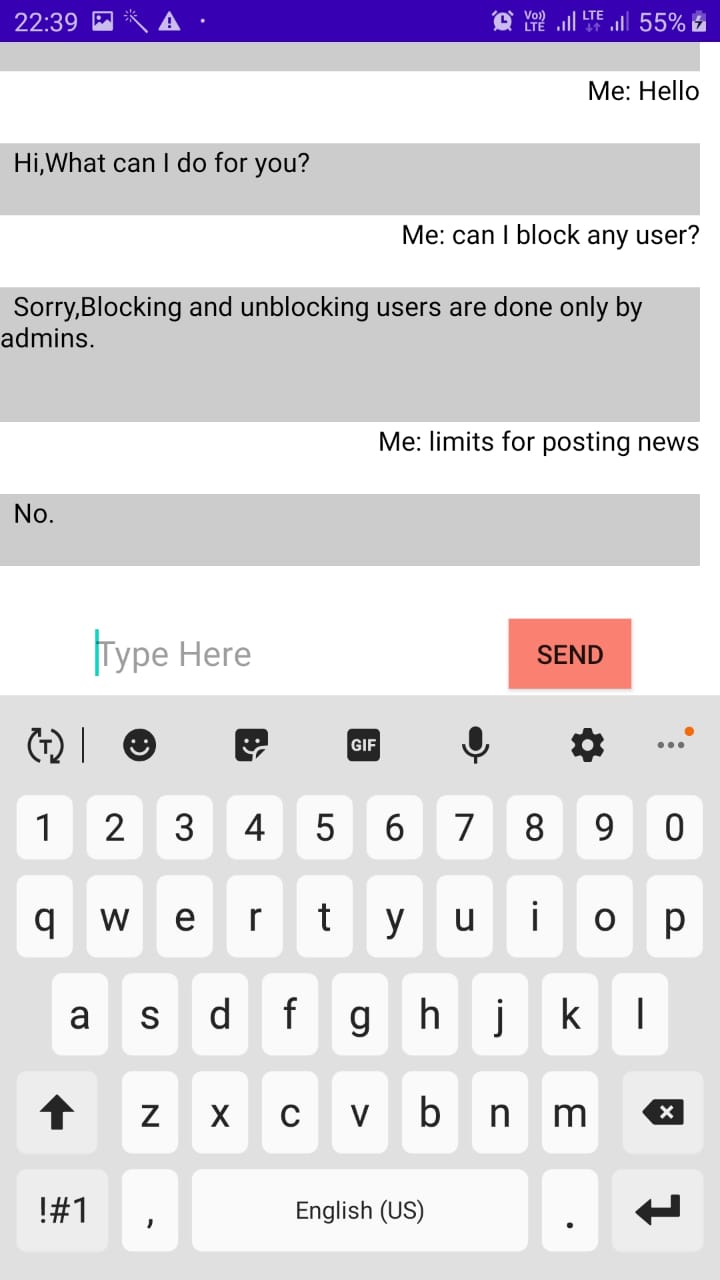
PAGE TO VIEW NEWS



SHARING NEWS



CHATBOT



**5.3.3 SAMPLE CODE**

@app.route('/newspost',methods=['post'])  
def addnews():  
 try:  
 heading=request.form['head']  
 content = request.form['cnt']  
 user\_id=request.form['uid']  
 result=checknews(content)  
 cmd.execute("select \* from newspost where heading='"+heading+"' and content='"+content+"' ")  
 s=cmd.fetchone()  
 if s is None:  
  
 cmd.execute("insert into newspost values(NULL,'"+user\_id+"','"+heading+"','"+content+"',curdate(),'"+result+"')")  
 con.commit()  
 return jsonify({'task': "success"})  
 else:  
 return jsonify({'task': "unsuccess"})  
 except Exception as e:  
 print(str(e))  
 return jsonify({'task': "Failed"})

def checknews(news):  
  
  
 res = stop(news)  
 key = ''  
 i = 1  
 resultset = []  
 resultset1 = []  
  
 for r in res:  
 key = key + " " + r  
 keyval = key  
  
 print('https://www.google.co.in/search?rlz=1C1CHBF\_enIN790IN790&biw=935&bih=657&ei=CDVcXLOCJJeRwgPorKjwDA&q=' + keyval)  
  
 res1 = requests.get(  
 'https://www.google.co.in/search?rlz=1C1CHBF\_enIN790IN790&biw=935&bih=657&ei=CDVcXLOCJJeRwgPorKjwDA&q=' + keyval)  
 # print(res1.text)  
 resn = []  
  
 # print(res1.text)  
 import re  
 clean = re.compile('<.\*?>')  
  
 # print(res1.text)  
 # class="ZINbbc xpd O9g5cc uUPGi"  
 ll = res1.text.split('<div class="ZINbbc xpd O9g5cc uUPGi"')  
 # print(ll)  
 #  
 print(len(ll), "lennnnnnnnnnnnnnnnnnnnnnnnn")  
 for i in range(1, len(ll)-1):  
 ll1 = ll[i]  
 print(type(ll1))  
 lll=ll1.split('<div class="kCrYT">')  
 if len(lll)>2:  
  
 newsl=lll[2]  
 print(newsl)  
 print("==============================================================")  
 text = re.sub(clean, "", newsl)  
 print(text)  
 print("==============================================================")  
  
 resn.append(text)  
 # print(resn, " urlsssssssssssssssssss")  
  
  
 sim = []  
 for n in resn:  
 # print(n)  
  
 dictl = process(n)  
  
 print("dictl",n,dictl)  
  
 dict2 = process(news)  
 print("dict2",news,dict2)  
  
 sim.append(getsimilarity(dict2, dictl))  
 print("=======================================")  
 #  
 print("similarity between Bug#599831 and Bug#800279 is ", sim)  
 sum = 0.0  
 cou = 0  
 print("sim",sim)  
 for s in sim:  
 if float(s) > 0.45:  
 cou = cou + 1  
 sum = sum + float(s)  
  
 sum = sum / len(sim)  
 conn = cou / len(sim)  
 print(cou / len(sim))  
 print(sum)  
 thr = ""  
 if conn >= 0.5:  
 thr = "real"  
 else:  
 thr = "fake"  
 # cmd.execute("insert into news values(null,'" + str(uid) + "','" + heading + "','" + news + "',curdate(),'"+thr+"')")  
 # con.commit()  
 print(thr)  
 return thr  
  
  
def stop(text):  
  
 from nltk.corpus import stopwords  
 from nltk.tokenize import word\_tokenize  
 import numpy as np  
 import nltk  
  
 def process(file):  
 raw = open(file).read()  
 tokens = word\_tokenize(raw)  
 words = [w.lower() for w in tokens]  
 porter = nltk.PorterStemmer()  
 stemmed\_tokens = [porter.stem(t) for t in words]  
 # Removing stop words  
 stop\_words = set(stopwords.words('english'))  
 filtered\_tokens = [w for w in stemmed\_tokens if not w in stop\_words]  
 # count words  
 count = nltk.defaultdict(int)  
 for word in filtered\_tokens:  
 count[word] += 1  
 return count;  
  
 def cos\_sim(a, b):  
 dot\_product = np.dot(a, b)  
 norm\_a = np.linalg.norm(a)  
 norm\_b = np.linalg.norm(b)  
 return dot\_product / (norm\_a \* norm\_b)  
  
 def getsimilarity(dictl, dict2):  
 all\_words\_list = []  
 for key in dictl:  
 all\_words\_list.append(key)  
 for key in dict2:  
 all\_words\_list.append(key)  
 all\_words\_list\_size = len(all\_words\_list)  
  
 v1 = np.zeros(all\_words\_list\_size, dtype=np.int)  
 v2 = np.zeros(all\_words\_list\_size, dtype=np.int)  
 i = 0  
 for (key) in all\_words\_list:  
 v1[i] = dictl.get(key, 0)  
 v2[i] = dict2.get(key, 0)  
 i = i + 1  
 return cos\_sim(v1, v2)  
  
 example\_sent = text.lower()  
 example\_sent=str(example\_sent).replace('-',' ')  
 example\_sent = str(example\_sent).replace('\_', ' ')  
 stop\_words= set(stopwords.words('english'))  
 word\_tokens= word\_tokenize(example\_sent)  
  
 filtered\_sentence= [w for w in word\_tokens if not w in stop\_words]  
  
 filtered\_sentence=[]  
  
 for w in word\_tokens:  
 if w not in stop\_words:  
 filtered\_sentence.append(w)  
  
  
 return filtered\_sentence  
def process(file):  
 raw =file  
 tokens = word\_tokenize(raw)  
 words = [w. lower() for w in tokens]  
 porter = nltk.PorterStemmer()  
 stemmed\_tokens = [porter. stem(t) for t in words]  
 # Removing stop words  
 stop\_words = set(stopwords.words( 'english' ) )  
 filtered\_tokens = [w for w in stemmed\_tokens if not w in stop\_words]  
 # count words  
 count = nltk.defaultdict(int)  
 for word in filtered\_tokens:  
 count [word] += 1  
 return count;  
  
def cos\_sim(a, b):  
 dot\_product = np.dot(a, b)  
 norm\_a = np.linalg.norm(a)  
 norm\_b = np.linalg.norm(b)  
 return dot\_product / (norm\_a \* norm\_b)  
  
def getsimilarity(dictl, dict2) :  
 all\_words\_list= []  
 for key in dictl:  
 all\_words\_list.append(key)  
 for key in dict2:  
 all\_words\_list.append(key)  
 all\_words\_list\_size = len(all\_words\_list)  
  
 v1 = np.zeros(all\_words\_list\_size, dtype=np.int)  
 v2 = np.zeros(all\_words\_list\_size, dtype=np.int)  
 i = 0  
 for (key) in all\_words\_list:  
 v1[i] = dictl.get(key, 0)  
 v2[i] = dict2.get(key, 0)  
 i = i+1  
 return cos\_sim(v1, v2)  
  
  
  
  
  
if \_\_name\_\_=="\_\_main\_\_":  
 # checknews("farmers strike came to end")  
  
 app.run(host='192.168.43.29',port=5000)