A

Major Project

On

DETECTING AND REMOVING MALICIOUS SOCIAL BOTS

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY

BY

T.VIJAYA LAXMI (177R1A1246)
A.SAHANA REDDY (177R1A1203)
B.DEVIKA (177R1A1202)

Under the Guidance of

Dr. M.VARAPRASAD RAO

(Professor)

DEPARTMENT OF INFORMATION TECHNOLOGY

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)

Recognized Under Section 2(f) & 12(B) of the UGCAct.1956,

Kandlakoya (V), Medchal Road, Hyderabad-501401.

2017-21

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled "**DETECTING AND REMOVING MALICIOUS SOCIAL BOTS**" being submitted by **T.VIJAYA LAXMI(177R1A1246),A.SAHANA REDDY(177R1A1203), B.DEVIKA(177R1A1202)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Information Technology of the Jawaharlal Nehru Technological University Hyderabad, during the year 2020-2021.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr. M. Varaprasad Rao
Professor
INTERNAL GUIDE

Dr. A. RajiReddy
DIRECTOR

HOD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on _____

ACKNOWLEDGEMENT

opportunity to express a deep sense of gratitude to Project Review Committee (PRC) Coordinators: **Mr. J. Narasimha Rao, Mr. B. P. Deepak Kumar, Mr. K. Murali, Dr. Suwarna Gothane and Mr. B. Ramji** for their cordial support, valuable information and guidance Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to our guide **Dr. M.Varaprasad Rao**, Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this, which helped us in completing this task through various stages.

We are also thankful to Head, Department of Information Technology for providing encouragement and support for completing this project successfully.

We are obliged to our Director **Dr. A. Raji Reddy** for being cooperative throughout the course of this project. We would like to express our sincere gratitude to our Chairman Sri. **Ch. Gopal Reddy** for his encouragement throughout the course of this project.

The guidance and support received from all the members of **CMR TECHNICAL CAMPUS** who contributed and who are contributing to this project, was vital for the success of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity thank our family for their constant encouragement without which this assignment would not be possible. We sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project.

T.VIJAYA LAXMI (177R1A1246)
A.SAHANA REDDY (177R1A1203)
B.DEVIKA (177R1A1202)

ABSTRACT

With the significant increase in the volume, velocity, and variety of user data (e.g., user generated data) in online social networks, there have been attempted to design new ways of collecting and analyzing such big data. For example, social bots have been used to perform automated analytical services and provide users with improved quality of service. However, malicious social bots have also been used to disseminate false information (e.g., fake news), and this can result in real-world consequences. Therefore, detecting and removing malicious social bots in online social networks is crucial. The most existing detection methods of malicious social bots analyze the quantitative features of their behavior. These features are easily imitated by social bots; thereby resulting in low accuracy of the analysis. A novel method of detecting malicious social bots, including both features selection based on the transition probability of clickstream sequences and semi-supervised clustering, is presented in this paper. This method not only analyzes transition probability of user behavior clickstreams but also considers the time feature of behavior. Findings from our experiments on real online social network platforms demonstrate that the detection accuracy for different types of malicious social bots by the detection method of malicious social bots based on transition probability of user behavior clickstreams increases by an average of 12.8%, in comparison to the detection method based on quantitative analysis of user behavior.

LIST OF FIGURES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1.INTRODUCTION

1.INTRODUCTION AND OBJECTIVES

1.1 Introduction:

In online social networks, social bots are social accounts controlled by automated programs that can perform corresponding operations based on a set of procedures. The increasing use of mobile devices (e.g., Android and iOS devices) also contributed to an increase in the frequency and nature of user interaction via social networks. It is evidenced by the significant volume, velocity and variety of data generated from the large online social network user base. Social bots have been widely deployed to enhance the quality and efficiency of collecting and analyzing data from social network services. However, public opinion about social networks and massive user data can also be mined or disseminated for malicious or nefarious purpose. In online social networks, automatic social bots cannot represent the real desires and intentions of normal human beings, so they are usually looked upon malicious one. In previous research, various methods were used to protect the security of online social network. User behavior is the most direct manifestation of user intent, as different users have different habits, preferences, and online behavior. Detect malicious social bots, and reduce the harm of malicious social bots, we need to acquire and analyze social situation of user behavior and compare and understand the differences of malicious social bots and normal users in dynamic behavior.

1.2 Purpose of the project:

Specifically, we aim to detect malicious social bots on social network platforms in real-time, by (1) proposing the transition probability features between user clickstreams based on the social situation analytics; and (2) designing an algorithm for detecting malicious social bots based on spatiotemporal features.

1.3 Existing systems:

The most existing detection methods of malicious social bots analyze the quantitative features of their behavior. These features are easily imitated by social bots; thereby resulting in low accuracy of the analysis. A novel method of detecting malicious social bots, including both features selection based on the transition probability of clickstream sequences and semi-supervised clustering, is presented.

1.4 Proposed system:

Proposed that situation analytics can be included in software service requirement analysis, which can facilitate the analysis of any change in user's requirements. Such an analysis is useful to understand the dynamic needs of a software service environment.

Proposed a heuristic-type supervised Boost OR model with increasing recall rate to detect malicious bots, which using the proportion of tweets forwarded to the published tweets on the Twitter, the mean length of tweets, URL, and forwarding interval.

According to the social interactions between users of the Twitter user to identify the active, passive and inactive users, a supervised machine learning method was proposed to identify social bots on the basis of age, location and other static features of active, passive, and inactive users in the Twitter, as well as interacting person, interaction content, interaction theme, and some dynamic characteristics.

We then analyze and classify situation aware user behaviors in social networks using our proposed semi-supervised clustering detection method. This allows us to promptly detect malicious social bots using only a small number of tagged users.

2.LITERATURE SURVEY

2. LITERATURE SURVEY

H. Gao *et al.*, ``Spam ain't as diverse as it seems: Throttling OSN spam with templates underneath' For example, some fake social bots accounts created to imitate the profile of a normal user, steal user data and compromise their privacy [4], disseminate malicious or fake information [5], [6], malicious comment, promote or advance certain political or ideology agenda and propaganda [7], and influence the stock market and other societal and economical markets [8]

M. Al-Qurishi, M. S. Hossain, M. Alrubaian, S. M. M. Rahman, and A. Alamri, ``Leveraging analysis of user behavior to identify malicious activities in large-scale social networks. collected a large amount of user information on the Twitter and YouTube, about 13 million channel activities, analyzing and detecting abnormal behaviors that deviate significantly from large-scale specifications through user behavior in two social networks

E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, ``The rise of social bots,"
For example, some fake social bots accounts created to imitate the profile of a normal user, steal user data and compromise their privacy , disseminate malicious or fake information, malicious comment, promote or advance certain political or ideology agenda and propaganda, and influence the stock market and other societal and economical markets .

F. Morstatter, L. Wu, T. H. Nazer, K. M. Carley, and H. Liu, ``A new approach to bot detection: Striking the balance between precision and recall,". proposed a heuristic-type supervised BoostOR model with increasing recall rate to detect malicious bots, which using the proportion of tweets forwarded to the published tweets on the Twitter, the mean length of tweets, URL, and forwarding interval.

3. SOFTWARE REQUIREMNT ANALYSIS

3. SOFTWARE REQUIREMENT ANALYSIS

3.1 PROBLEM SPECIFICATION:

Client Server

Over view:

With the varied topic in existence in the fields of computers, Client Server is one, which has generated more heat than light, and also more hype than reality. This technology has acquired a certain critical mass attention with its dedication conferences and magazines. Major computer vendors such as IBM and DEC, have declared that Client Servers is their main future market. A survey of DBMS magazine revealed that 76% of its readers were actively looking at the client server solution. The growth in the client server development tools from $200 million in 1992 to more than $1.2 billion in 1996.

What is a Client Server?

Two prominent systems in existence are client server and file server systems. It is essential to distinguish between client servers and file server systems. Both provide shared network access to data but the comparison dens there! The file server simply provides a remote disk drive that can be accessed by LAN applications on a file by file basis.

Why Client Server:

Client server has evolved to solve a problem that has been around since the earliest days of computing: how best to distribute your computing, data generation and data storage resources in order to obtain efficient, cost effective departmental and enterprise wide data processing. During mainframe era choices were quite limited. A central machine housed both the CPU and DATA (cards, tapes, drums and later disks). A strong central information service department ruled the corporation. The role of the rest of the corporation limited to requesting new or more frequent reports and to provide hand written forms from which the central data banks were created and updated. The earliest client server solutions therefore could best be characterized as “SLAVE-MASTER”. Time-sharing changed the picture. Remote terminal could view and even change the central data, subject to access permissions. And, as the central data banks evolved in to sophisticated relational database with non-programmer query languages, online users could formulate ad hoc queries and produce local reports without adding to the MIS applications software backlog. However remote access was through dumb terminals, and the client server remained subordinate to the Slave\Master.

Front end and user interface design:

The entire user interface is planned to be developed in browser specific environment with a touch of Intranet-Based Architecture for achieving the Distributed Concept. The browser specific components are designed by using the HTML standards, and the dynamism of the designed by concentrating on the constructs of the Java Server Pages.

Communication or Database Connectivity Tier:

The Communication architecture is designed by concentrating on the Standards of Servlets and Enterprise Java Beans. The database connectivity is established by using the Java Data Base Connectivity. The standards of three-tier architecture are given major concentration to keep the standards of higher cohesion and limited coupling for effectiveness of the operations.

Features of the Language Used:

In my project, I have chosen **Java** language for developing the code.

About Java:

Initially the language was called as “oak” but it was renamed as “Java” in 1995. The primary motivation of this language was the need for a platform-independent (i.e., architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

- Java is a programmer’s language.
- Java is cohesive and consistent.
- Except for those constraints imposed by the Internet environment, Java gives the programmer, full control.

Finally, Java is to Internet programming where C was to system programming.

Importance of java on internet:

Java has had a profound effect on the Internet. This is because; Java expands the Universe of objects that can move about freely in Cyberspace. The Dynamic, Self-executing programs cause serious problems in the areas of Security and probability. But, Java addresses those concerns and by doing so, has opened the door to an exciting new form of program called the Applet.

Java can be used to create two types of programs:

***Applications* and *Applets*:** An application is a program that runs on our Computer under the operating system of that computer. It is more or less like one creating using C or C++. Java's ability to create Applets makes it important. An Applet is an application designed to be transmitted over the Internet and executed by a Java –compatible web browser. An applet is actually a tiny Java program, dynamically downloaded across the network, just like an image. But the difference is, it is an intelligent program, not just a media file. It can react to the user input and dynamically change.

Features Of Java

Security:

Every time you that you download a "normal" program; you are risking a viral infection. Prior to Java, most users did not download executable programs frequently, and those who did scan them for viruses prior to execution. Most users still worried about the possibility of infecting their systems with a virus. In addition, another type of malicious program exists that must be guarded against. This type of program can gather private information, such as credit card numbers, bank account balances, and passwords. Java answers both these concerns by providing a "firewall" between a network application and your computer.

When you use a Java-compatible Web browser, you can safely download Java applets without fear of virus infection or malicious intent.

Portability:

For programs to be dynamically downloaded to all the various types of platforms connected to the Internet, some means of generating portable executable code is needed .As you will see, the same mechanism that helps ensure security also helps create portability. Indeed, Java's solution to these two problems is both elegant and efficient.

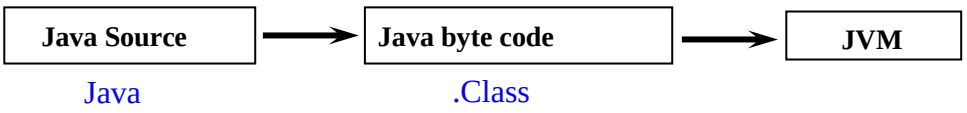
The Byte code:

The key that allows the Java to solve the security and portability problems is that the output of Java compiler is Byte code. Byte code is a highly optimized set of instructions designed to be executed by the Java run-time system, which is called the Java Virtual Machine (JVM). That is, in its standard form, the JVM is an interpreter for byte code.

Java Virtual Machine (JVM) :

Beyond the language, there is the Java virtual machine. The Java virtual machine is an important element of the Java technology. The virtual machine can be embedded within a web browser or an operating system. Once a piece of Java code is loaded onto a machine, it is verified. As part of the loading process, a class loader is invoked and does byte code verification makes sure that the code that's has been generated by the compiler will not corrupt the machine that it's loaded on. Byte code verification takes place at the end of the compilation process to make sure that is all accurate and correct. So byte code verification is integral to the compiling and executing of Java code.

OVER ALL DEESCRIPTION:



Java Architecture:

Java architecture provides a portable, robust, high performing environment for development. Java provides portability by compiling the byte codes for the Java Virtual Machine, which is then interpreted on each platform by the run-time environment. Java is a dynamic system, able to load code when needed from a machine in the same room or across the planet.

Compilation of code:

When you compile the code, the Java compiler creates machine code (called byte code) for a hypothetical machine called Java Virtual Machine (JVM). The JVM is supposed to execute the byte code. The JVM is created for overcoming the issue of portability. The code is written and compiled for one machine and interpreted on all machines.

Simple:

Java was designed to be easy for the Professional programmer to learn and to use effectively. If you are an experienced C++ programmer, learning Java will be even easier. Because Java inherits the C/C++ syntax and many of the objects oriented features of C++. Most of the confusing concepts from C++ are either left out of Java or implemented in a cleaner, more approachable manner. In Java there are a small number of clearly defined ways to accomplish a given task.

Object-Oriented:

Java was not designed to be source-code compatible with any other language. This allowed the Java team the freedom to design with a blank slate. One outcome of this was a clean usable, pragmatic approach to objects. The object model in Java is simple and easy to extend, while simple types, such as integers, are kept as high-performance non-objects.

Robust:

The multi-platform environment of the Web places extraordinary demands on a program, because the program must execute reliably in a variety of systems. The ability to create robust programs was given a high priority in the design of Java. Java is strictly typed language; it checks your code at compile time and run time.

Java virtually eliminates the problems of memory management and DE allocation, which is completely automatic. In a well-written Java program, all run time errors can –and should –be managed by your program.

JAVASCRIPT:

JavaScript is a script-based programming language that was developed by Netscape Communication Corporation. JavaScript was originally called Live Script and renamed as JavaScript to indicate its relationship with Java. JavaScript supports the development of both client and server components of Web-based applications. On the client side, it can be used to write programs that are executed by a Web browser within the context of a Web page. On the server side, it can be used to write Web server programs that can process information submitted by a Web browser and then update the browser's display accordingly

<SCRIPTS>..</SCRIPT>.

<SCRIPT LANGUAGE = "JavaScript">

JavaScript statements

</SCRIPT>

Here are a few things we can do with JavaScript:

- Validate the contents of a form and make calculations.
- Add scrolling or changing messages to the Browser's status line.
- Animate images or rotate images that change when we move the mouse over them.
- Detect the browser in use and display different content for different browsers.
- Detect installed plug-ins and notify the user if a plug-in is required.

We can do much more with JavaScript, including creating entire application.

JavaScript vs. java:

JavaScript and Java are entirely different languages. A few of the most glaring differences are:

- Java applets are generally displayed in a box within the web document; JavaScript can affect any part of the Web document itself.
- While JavaScript is best suited to simple applications and adding interactive features to Web pages; Java can be used for incredibly complex applications.

There are many other differences but the important thing to remember is that JavaScript and Java are separate languages. They are both useful for different things; in fact they can be used together to combine their advantages.

Advantages:

- JavaScript can be used for Sever-side and Client-side scripting.
- It is more flexible than VBScript.
- JavaScript is the default scripting languages at Client-side since all the browsers supports it.

Hyper Text Markup Language:

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows users to produces Web pages that include text, graphics and pointer to other Web pages (Hyperlinks).

HTML is not a programming language but it is an application of ISO Standard 8879, SGML (Standard Generalized Markup Language), but specialized to hypertext and adapted to the Web. The idea behind Hypertext is that instead of reading text in rigid linear structure, we can easily jump from one point to another point. We can navigate through the information based on our interest and preference. A markup language is simply a series of elements, each delimited with special characters that define how text or other items enclosed within the elements should be displayed. Hyperlinks are underlined or emphasized works that load to other documents or some portions of the same document.

HTML can be used to display any type of document on the host computer, which can be geographically at a different location. It is a versatile language and can be used on any platform or desktop.

HTML provides tags (special codes) to make the document look attractive. HTML tags are not case-sensitive. Using graphics, fonts, different sizes, color, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

Basic HTML Tags :

<! -- -->	Specifies comments
<A>…….</A>	Creates hypertext links
<B>…….</B>	Formats text as bold
<BIG>…….</BIG>	Formats text in large font.
<BODY>…</BODY>	Contains all tags and text in the HTML document
<CENTER>...</CENTER>	Creates text
<DD>…</DD>	Definition of a term
<DL>...</DL>	Creates definition list
<FONT>…</FONT>	Formats text with a particular font
<FORM>...</FORM>	Encloses a fill-out form
<FRAME>...</FRAME>	Defines a particular frame in a set of frames
<H#>…</H#>	Creates headings of different levels
<HEAD>...</HEAD>	Contains tags that specify information about a document
<HR>...</HR>	Creates a horizontal rule
<HTML>…</HTML>	Contains all other HTML tags
<META>...</META>	Provides meta-information about a document
<SCRIPT>…</SCRIPT>	Contains client-side or server-side script
<TABLE>…</TABLE>	Creates a table
<TD>…</TD>	Indicates table data in a table
<TR>…</TR>	Designates a table row
<TH>…</TH>	Creates a heading in a table

ADVANTAGES:

- A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.
- HTML is platform independent.
- HTML tags are not case-sensitive.

Java Database Connectivity

What Is JDBC?

JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for Java Database Connectivity. It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. The combinations of Java and JDBC lets a programmer write it once and run it anywhere.

What Does JDBC Do?

Simply put, JDBC makes it possible to do three things:

- Establish a connection with a database
- Send SQL statements
- Process the results.

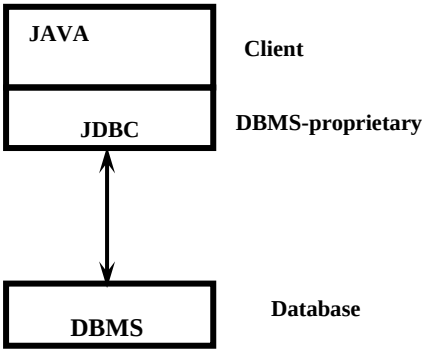
JDBC VERSUS ODBS AND OTHER APIS

So why not just use ODBC from Java? The answer is that you can use ODBC from Java, but this is best done with the help of JDBC in the form of the JDBC-ODBC Bridge, which we will cover shortly. The question now becomes "Why do you need JDBC?" There are several answers to this question:

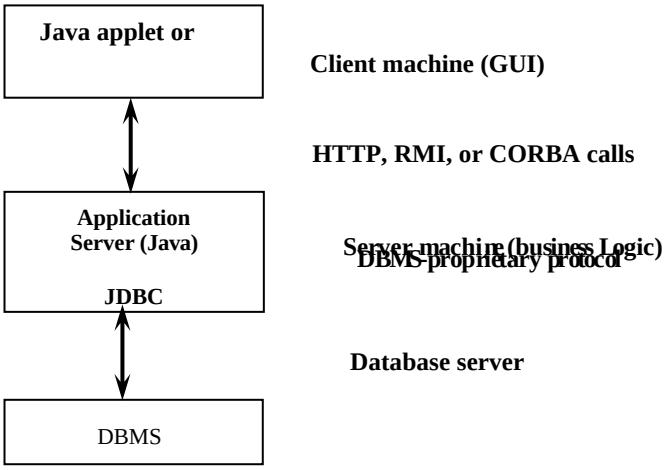
1. ODBC is not appropriate for direct use from Java because it uses a C interface. Calls from Java to native C code have a number of drawbacks in the security, implementation, robustness, and automatic portability of applications.
2. A literal translation of the ODBC C API into a Java API would not be desirable. For example, Java has no pointers, and ODBC makes copious use of them, including the notoriously error-prone generic pointer "void *". You can think of JDBC as ODBC translated into an object-oriented interface that is natural for Java programmers.
3. ODBC is hard to learn. It mixes simple and advanced features together, and it has complex options even for simple queries. JDBC, on the other hand, was designed to keep simple things simple while allowing more advanced capabilities where required.
4. A Java API like JDBC is needed in order to enable a "pure Java" solution. When ODBC is used, the ODBC driver manager and drivers must be manually installed on every client machine. When the JDBC driver is written completely in Java, however, JDBC code is automatically installable, portable, and secure on all Java platforms from network computers to mainframes.

Two tier and three tier models:

The JDBC API supports both two-tier and three-tier models for database access.
In the two-tier model, a Java applet or application talks directly to the database. This requires a JDBC driver that can communicate with the particular database management system being accessed. A user's SQL statements are delivered to the database, and the results of those statements are sent back to the user. The database may be located on another machine to which the user is connected via a network. This is referred to as a client/server configuration, with the user's machine as the client, and the machine housing the database as the server. The network can be an Intranet, which, for example, connects employees within a corporation, or it can be the Internet.



In the three-tier model, commands are sent to a "middle tier" of services, which then send SQL statements to the database. The database processes the SQL statements and sends the results back to the middle tier, which then sends them to the user. MIS directors find the



Three-tier model very attractive because the middle tier makes it possible to maintain control

Over access and the kinds of updates that can be made to corporate data. Another advantage is that when there is a middle tier, the user can employ an easy-to-use higher-level API which is translated by the middle tier into the appropriate low-level calls. Finally, in many cases the three-tier architecture can provide performance advantages.

Until now the middle tier has typically been written in languages such as C or C++, which offer fast performance. However, with the introduction of optimizing compilers that translate Java byte code into efficient machine-specific code, it is becoming practical to implement the middle tier in Java. This is a big plus, making it possible to take advantage of Java's robustness, multithreading, and security features. JDBC is important to allow database access from a Java middle tier.

JDBC Driver Types:

The JDBC drivers that we are aware of at this time fit into one of four categories:

- JDBC-ODBC bridge plus ODBC driver
- Native-API partly-Java driver
- JDBC-Net pure Java driver
- Native-protocol pure Java driver

JDBC-ODBC Bridge:

If possible, use a Pure Java JDBC driver instead of the Bridge and an ODBC driver. This completely eliminates the client configuration required by ODBC. It also eliminates the potential that the Java VM could be corrupted by an error in the native code brought in by the Bridge (that is, the Bridge native library, the ODBC driver manager library, the ODBC driver library, and the database client library).

What Is the JDBC- ODBC Bridge?

The JDBC-ODBC Bridge is a JDBC driver, which implements JDBC operations by translating them into ODBC operations. To ODBC it appears as a normal application program. The Bridge implements JDBC for any database for which an ODBC driver is available. The Bridge is implemented as the sun.jdbc.odbc Java package and contains a native library used to access ODBC. The Bridge is a joint development of Intersolv and Java Soft.

Java Server Pages (JSP):

Java server Pages is a simple, yet powerful technology for creating and maintaining dynamic-content web pages. Based on the Java programming language, Java Server Pages offers proven portability, open standards, and a mature re-usable component model .The Java Server Pages architecture enables the separation of content generation from content presentation. This separation not eases maintenance headaches; it also allows web team members to focus on their areas of expertise. Now, web page designer can concentrate on layout, and web application designers on programming, with minimal concern about impacting each other's work.

Features of JSP

Portability:

Java Server Pages files can be run on any web server or web-enabled application server that provides support for them. Dubbed the JSP engine, this support involves recognition, translation, and management of the Java Server Page lifecycle and its interaction components.

Components:

It was mentioned earlier that the Java Server Pages architecture can include reusable Java components. The architecture also allows for the embedding of a scripting language directly into the Java Server Pages file. The components current supported include Java Beans, and Servlets.

Processing:

A Java Server Pages file is essentially an HTML document with JSP scripting or tags. The Java Server Pages file has a JSP extension to the server as a Java Server Pages file. Before the page is served, the Java Server Pages syntax is parsed and processed into a Servlet on the server side. The Servlet that is generated outputs real content in straight HTML for responding to the client.

Access Models:

A Java Server Pages file may be accessed in at least two different ways. A client's request comes directly into a Java Server Page. In this scenario, suppose the page accesses reusable

Java Bean components that perform particular well-defined computations like accessing a database. The result of the Beans computations, called result sets is stored within the Bean as properties. The page uses such Beans to generate dynamic content and present it back to the client

Steps in the execution of a JSP Application:

1. The client sends a request to the web server for a JSP file by giving the name of the JSP file within the form tag of a HTML page.

2. This request is transferred to the Java Webserver. At the server side Java Webserver receives the request and if it is a request for a JSP file server gives this request to the JSP engine.
3. JSP engine is program which can understand the tags of the JSP and then it converts those tags into a Servlet program and it is stored at the server side. This Servlet is loaded in the memory and then it is executed and the result is given back to the Java Webserver and then it is transferred back to the result is given back to the Java Webserver and then it is transferred back to the client.

JDBC connectivity:

The JDBC provides database-independent connectivity between the J2EE platform and a wide range of tabular data sources. JDBC technology allows an Application Component Provider to:

- Perform connection and authentication to a database server
- Manager transactions
- Move SQL statements to a database engine for preprocessing and execution
- Execute stored procedures
- Inspect and modify the results from Select statements.

.

TOMCAT 6.0 WEB SERVER

Tomcat is an open source web server developed by Apache Group. Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Web Servers like Apache Tomcat support only web components while an application server supports web components as well as business components (BEAs Web logic, is one of the popular application server).To develop a web application with JSP/servlet install any web server like JRun, Tomcat etc. to run your application.

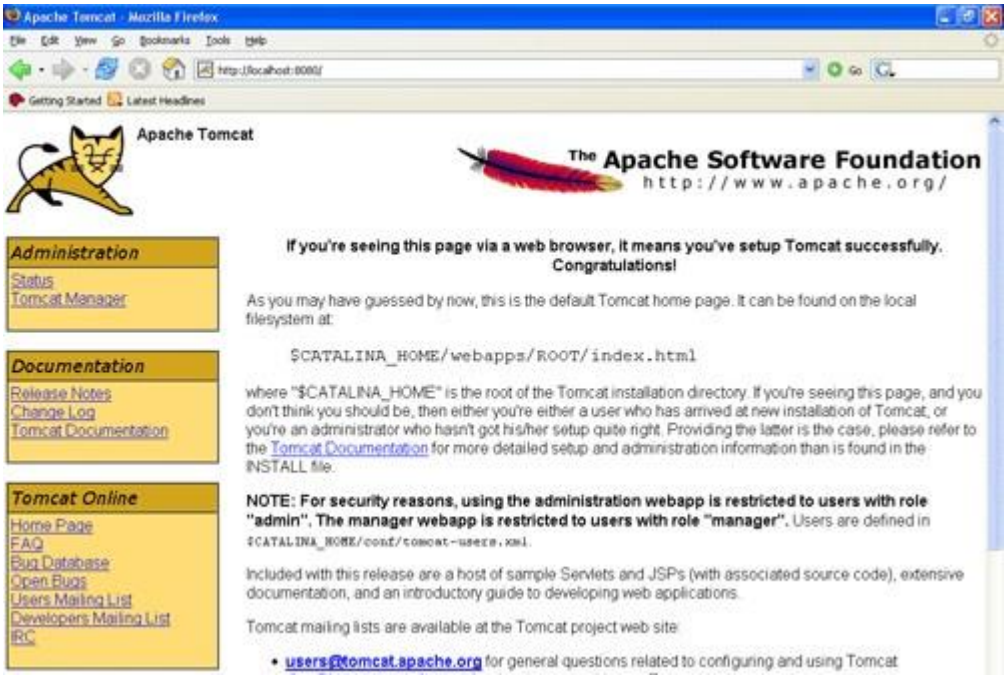


FIG 1: APACHE TOMACAT

Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.

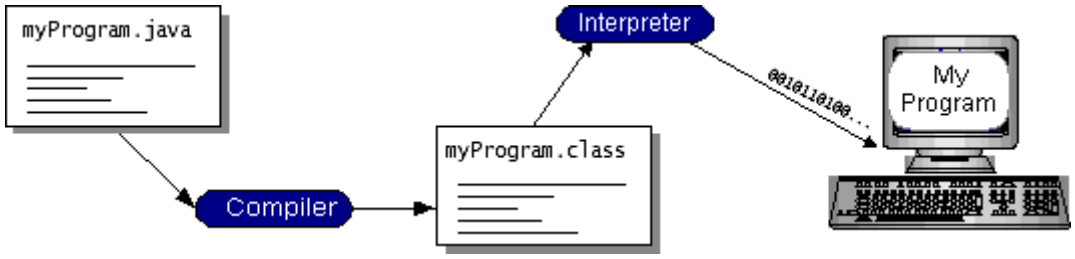


FIG 2: JAVA COMPILATION PROCESS

You can think of Java byte codes as the machine code instructions for the *Java Virtual Machine* (Java VM). Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make "write once, run anywhere" possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.

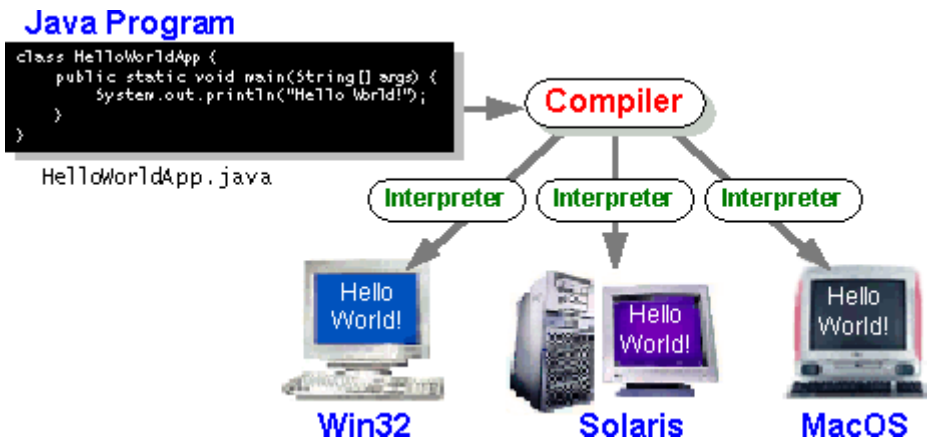


FIG 3 : JAVA COMPILER

ODBC:

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a *de facto*

Standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to.

JDBC:

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of Networking

TCP/IP stacks:

The TCP/IP stack is shorter than the OSI one: “plug-in” database connectivity modules

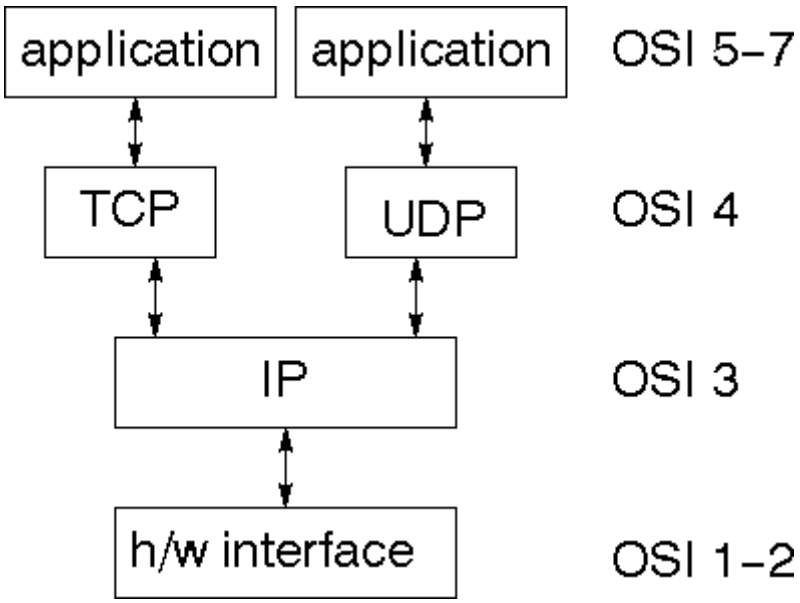
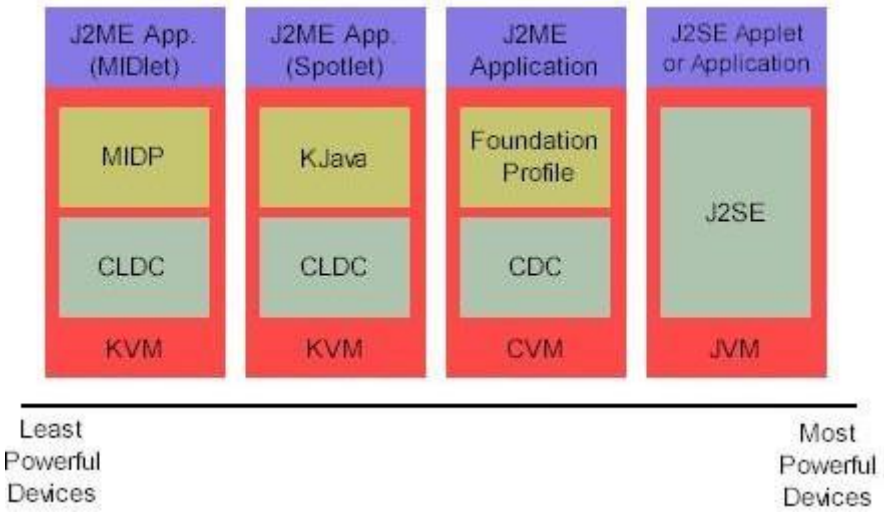


FIG 4: TCP/IP STRUCTURE

1. General J2ME architecture :



J2ME uses configurations and profiles to customize the Java Runtime Environment (JRE). As a complete JRE, J2ME is comprised of a configuration, which determines the JVM used, and a profile, which defines the application by adding domain-specific classes. About Java

Initially the language was called as “oak” but it was renamed as “Java” in 1995. The primary motivation of this language was the need for a platform-independent (i.e., architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

- Java is a programmer’s language.
- Java is cohesive and consistent.
- Except for those constraints imposed by the Internet environment, Java gives the programmer, full control.

Finally, Java is to Internet programming where C was to system programming.

Importance of Java to the Internet:

Java has had a profound effect on the Internet. This is because; Java expands the Universe of objects that can move about freely in Cyberspace. In a network, two categories of objects are transmitted between the Server and the Personal computer. They are: Passive information and Dynamic active programs. The Dynamic, Self-executing programs cause serious problems in the areas of Security and probability. But, Java addresses those concerns and by doing so, has opened the door to an exciting new form of program called the Applet.

Java can be used to two types of programs:

***Applications* and *Applets*:** An application is a program that runs on our Computer under the operating system of that computer. It is more or less like one creating using C or C++. Java's ability to create Applets makes it important. An Applet is an application designed to be transmitted over the Internet and executed by a Java –compatible web browser. An applet is actually a tiny Java program, dynamically downloaded across the network, just like an image. But the difference is, it is an intelligent program, not just a media file. It can react to the user input and dynamically change.

Security:

Every time you that you download a "normal" program; you are risking a viral infection. Prior to Java, most users did not download executable programs frequently, and those who did scan them for viruses prior to execution. Most users still worried about the possibility of infecting their systems with a virus. In addition, another type of malicious program exists that must be guarded against. This type of program can gather private information, such as credit card numbers, bank account balances, and passwords. Java answers both these concerns by providing a "firewall" between a network application and your computer.

When you use a Java-compatible Web browser, you can safely download Java applets without fear of virus infection or malicious intent.

Portability:

For programs to be dynamically downloaded to all the various types of platforms connected to the Internet, some means of generating portable executable code is needed .As you will see, the same mechanism that helps ensure security also helps create portability. Indeed, Java's solution to these two problems is both elegant and efficient.

The Byte code:

The key that allows the Java to solve the security and portability problems is that the output of Java compiler is Byte code. Byte code is a highly optimized set of instructions designed to

be executed by the Java run-time system, which is called the Java Virtual Machine (JVM). That is, in its standard form, the JVM is an interpreter for byte code.

Translating a Java program into byte code helps makes it much easier to run a program in a wide variety of environments. The reason is, once the run-time package exists for a given system, any Java program can run on it.

Although Java was designed for interpretation, there is technically nothing about Java that prevents on-the-fly compilation of byte code into native code. Sun has just completed its Just In Time (JIT) compiler for byte code. When the JIT compiler is a part of JVM, it compiles byte code into executable code in real time, on a piece-by-piece, demand basis. It is not possible to compile an entire Java program into executable code all at once, because Java performs various run-time checks that can be done only at run time. The JIT compiles code, as it is needed, during execution.

Java Virtual Machine (JVM):

Beyond the language, there is the Java virtual machine. The Java virtual machine is an important element of the Java technology. The virtual machine can be embedded within a web browser or an operating system. Once a piece of Java code is loaded onto a machine, it is verified. As part of the loading process, a class loader is invoked and does byte code verification makes sure that the code that's has been generated by the compiler will not corrupt the machine that it's loaded on. Byte code verification takes place at the end of the compilation process to make sure that is all accurate and correct. So byte code verification is integral to the compiling and executing of Java code.

3.2 MODULES AND THEIR FUNCTIONALITIES:

OSN Server:

In this module, the OSN Server has to login by using valid user name and password. After login successful he can do some operations such as view all user details and authorize them, list of all friends requests and response ,View all posts like images and messages user, view all Similar group users like doctors, Engineers, Business Man, etc.,. OSN Server can add some BOTS words to the database and view the all words added by him and based on that negative words admin can find all users behavior and also produce chart for that behavior words.

View and Authorize Users:

In this module, the admin can view the list of users who all registered. In this, the admin can view the user's details such as, user name, email, address and admin authorizes the users.

User:

In this module, there are n numbers of users are present. User should register with group option before doing some operations. After registration successful he has to wait for admin to authorize him and after admin authorized user can login by using authorized user name and password. Login successful he will do some operations like view profile details, Search friends based on keyword or friends name, view the friend requests, post message with image to all friends. Find posts of friends and comment on that posts.

Users can also view all his friends and delete those who don't want, view all group posts like doctor or engineer etc.,

Viewing Profile Details:

In this module, the user can see their own profile details, such as their address, email, mobile number, profile Image.

Search Friends, Request, and View Friend Requests, View all Friend Details:

In this, the user search for other users by their names, send requests and view friend requests from other users. User can see all his friend details with their images and personnel details.

3.3 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

Economic Feasibility:

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Technical Feasibility:

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

Social Feasibility:

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.4 FUNCTIONAL REQUIREMENTS

Input Design:

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.

Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases.

Output Design

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rests with the administrator only. The application starts running when it is executed for the first time. The server has to be started and then the internet explorer in used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

4. SOFTWARE AND HARDWARE REQUIREMENTS

4. SOFTWARE AND HARDWARE REQUIREMENTS:

4.1. HARDWARE REQUIREMENTS:

• System: Intel Dual Core.

• Hard Disk: 40 GB.

• RAM: 1 GB.

4.2 SOFTWARE REQUIREMENTS:

• Operating system: - Windows XP/7.

• Coding Language : JAVA/J2EE

• Data Base : MYSQL

• IDE :Net beans 8.1

• Server :Apache Tomcat

5.SOFTWARE DESIGN

5.SOFTWARE DESIGN

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

5.1 CLASS DIAGRAM:

The class diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed.

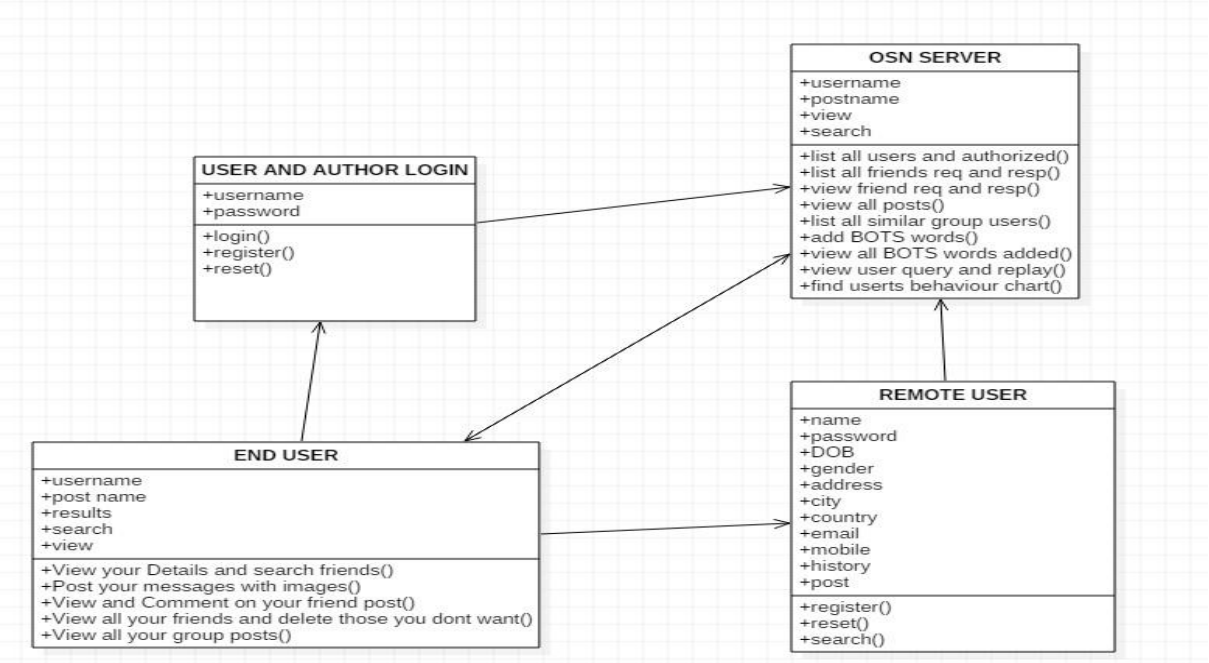


FIG 5: CLASS DIAGRAM

5.2 DATA FLOW DIAGRAM:

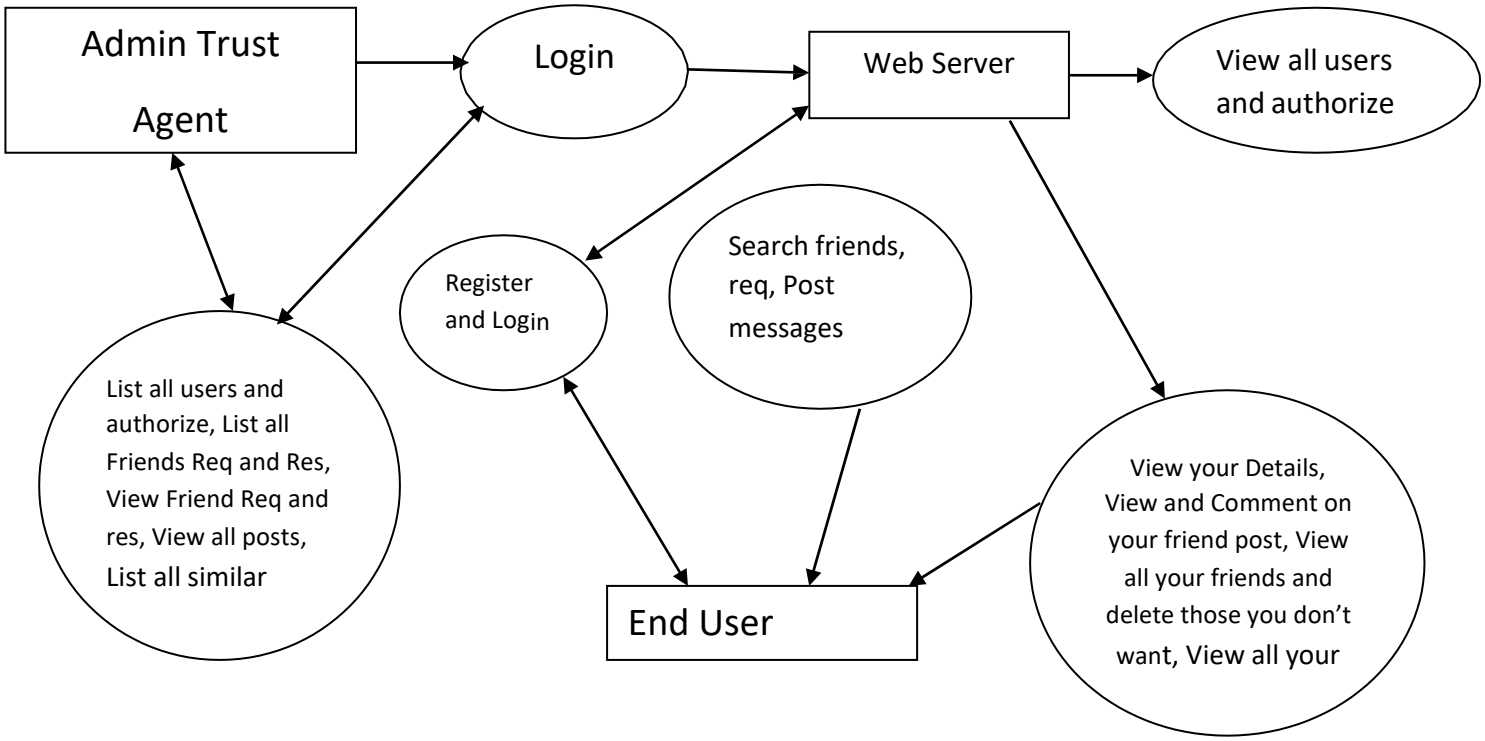


FIG 6: DATA FLOW DIAGRAM

5.3 USE CASE DIAGRAM:

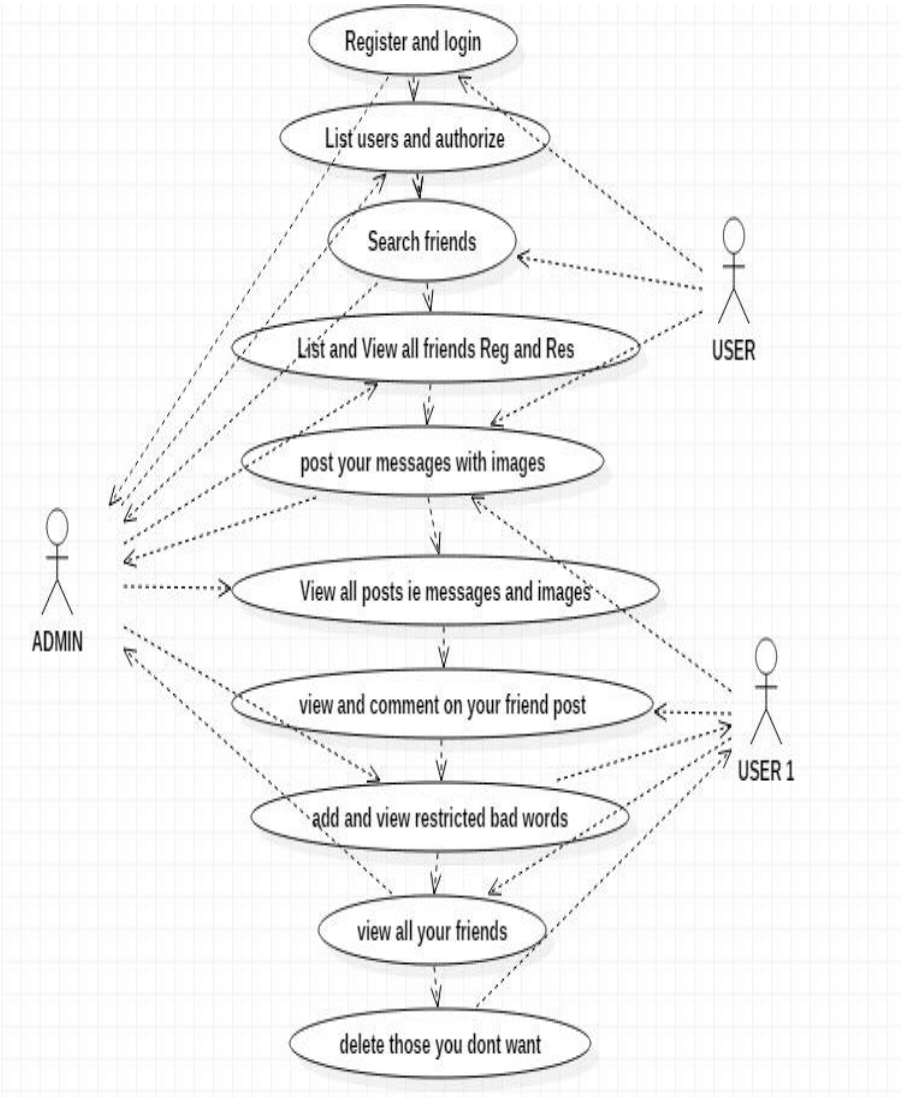


FIG 7: USE CASE DIAGRAM

5.4.1. FLOW CHART : ADMIN

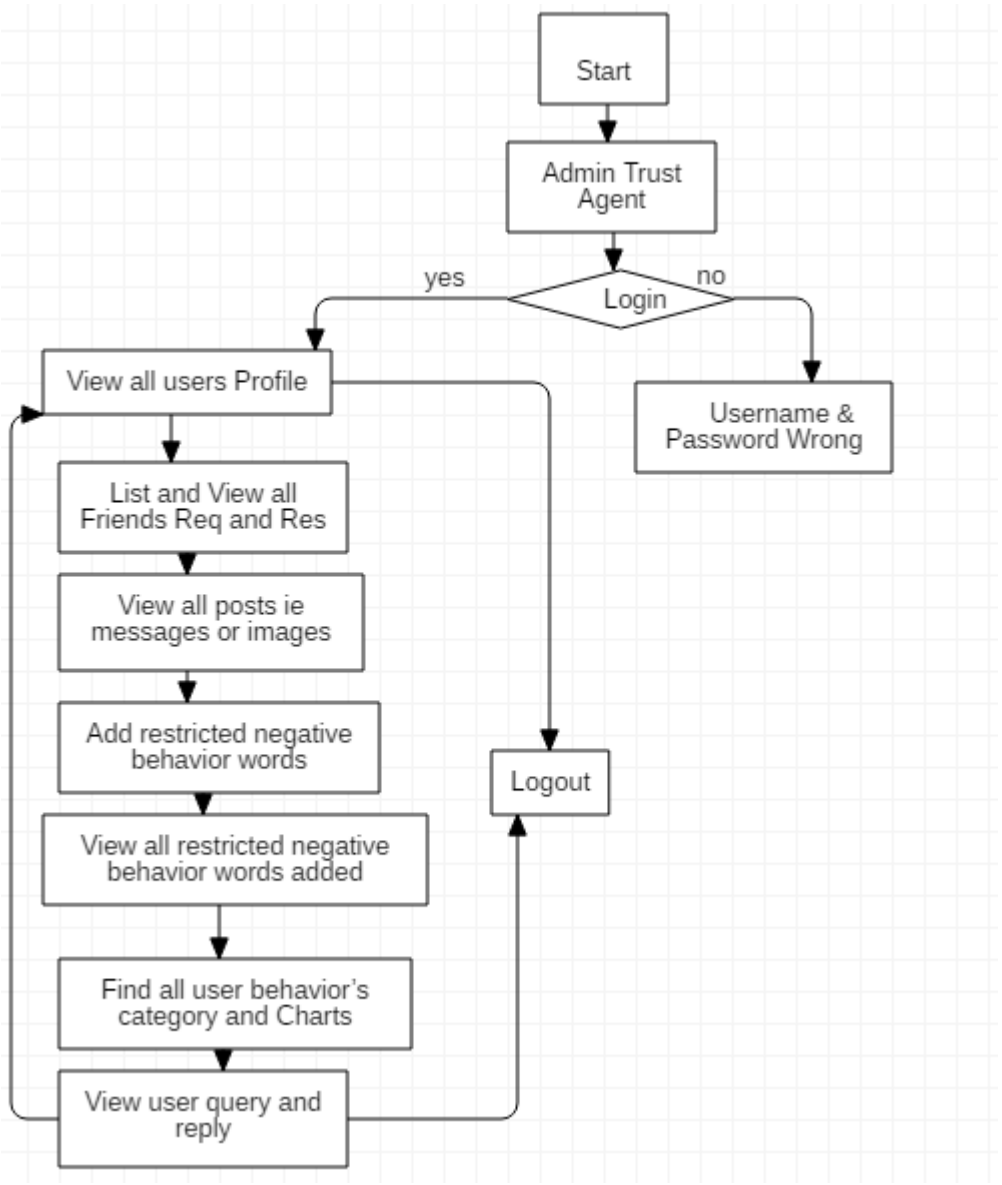


FIG 8: FLOW CHART ADMIN

5.4.2. FLOW CHART : USER

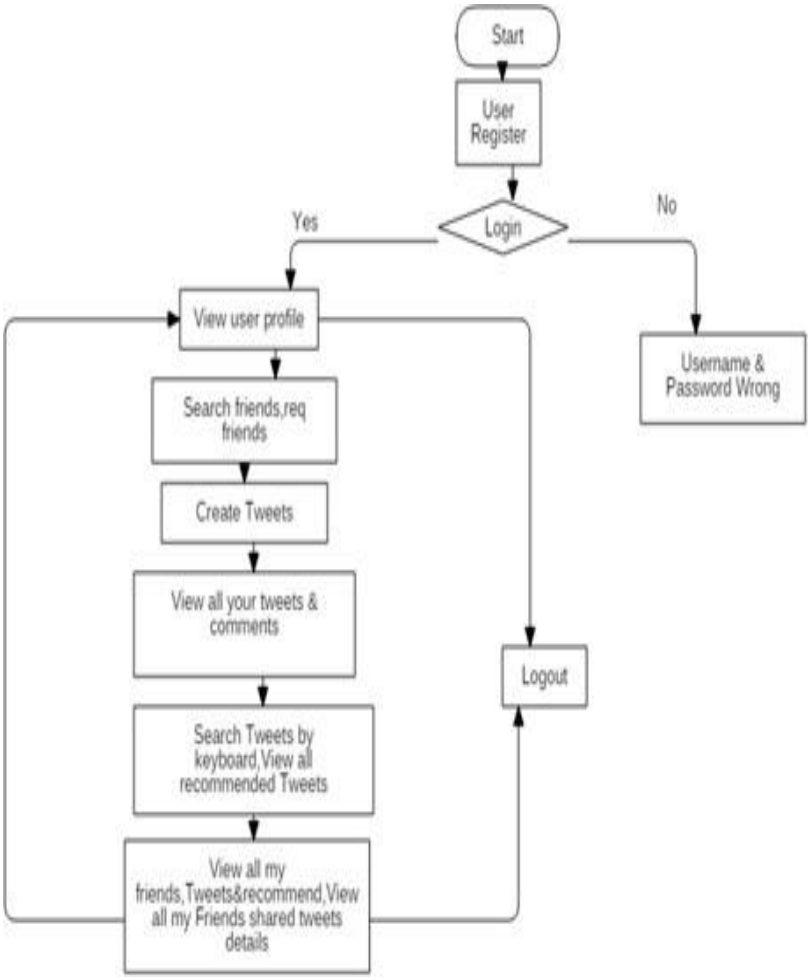


FIG 8.1: FLOW CHART USER

5.5 SEQUENCE DIAGRAM :

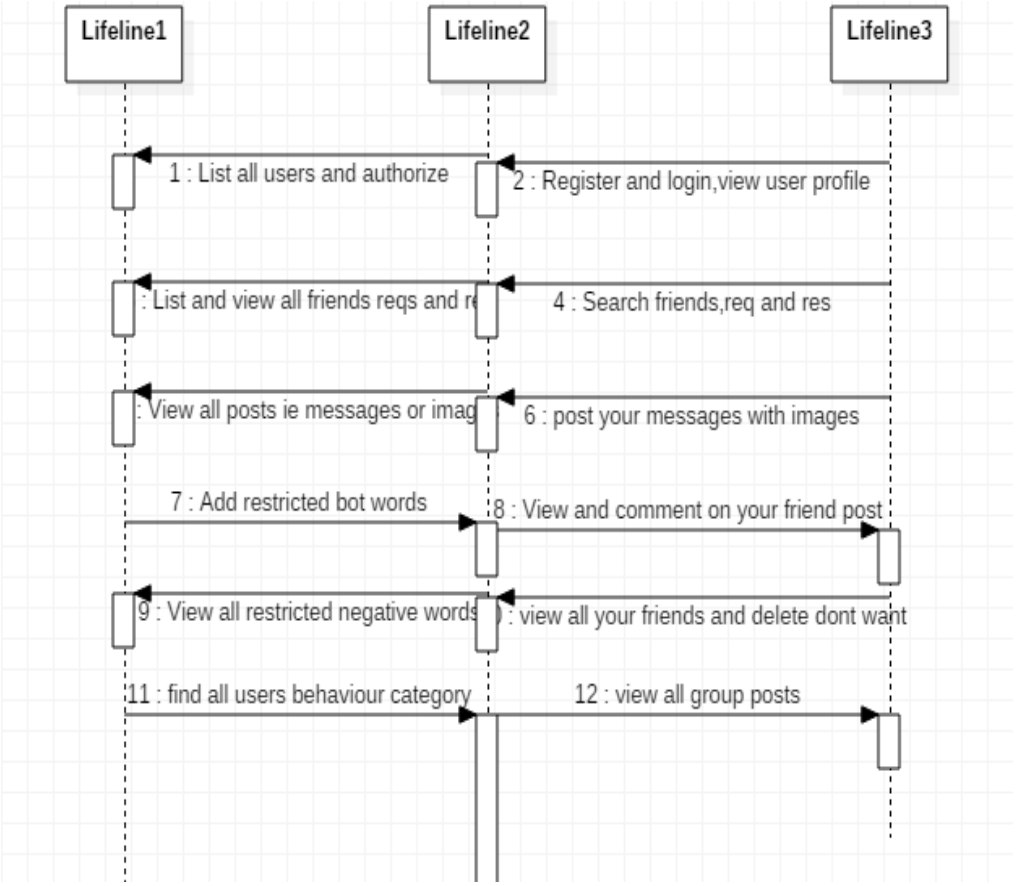


FIG 9: SEQUENCE DIAGRAM

5.6 ER DIAGRAM :

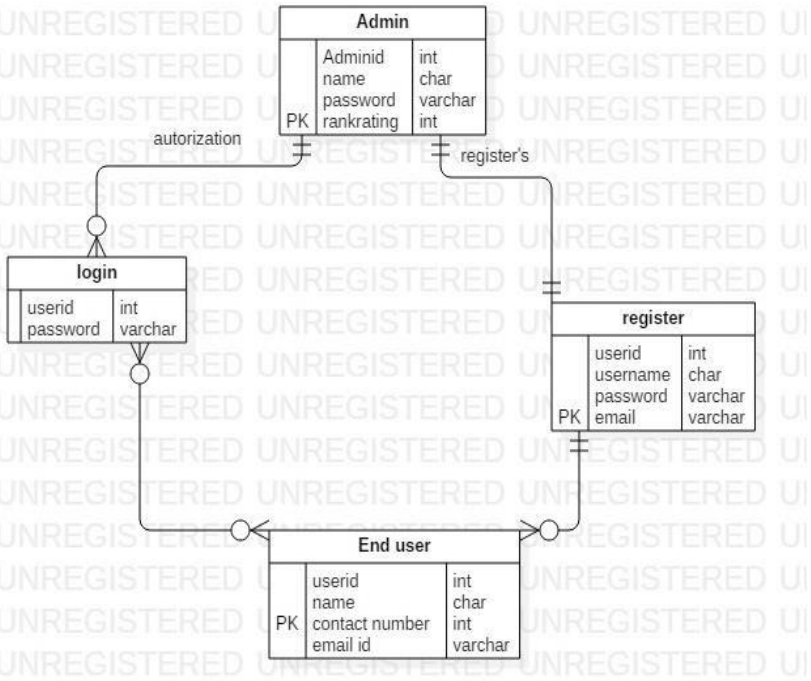


FIG 10: ER DIAGRAM

6. CODING AND IMPLEMENTATION

6. CODING AND IMPLEMENTATION

6.1 SAMPLE CODE

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<title>Home</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="style.css" rel="stylesheet" type="text/css" />

<script src="js/jquery-1.3.2.min.js" type="text/javascript"></script>

<script src="js/cufon-yui.js" type="text/javascript"></script>

<script src="js/cufon-replace.js" type="text/javascript"></script>

<script src="js/Myriad_Pro_400.font.js" type="text/javascript"></script>

<!--[if lt IE 7]>

<link href="ie_style.css" rel="stylesheet" type="text/css" />

<script type="text/javascript" src="js/ie_png.js"></script>

<script type="text/javascript">ie_png.fix('.png, .extra-bg, .box, .box1, .box2, .img-
list img');</script>

<![endif]-->

<style type="text/css">

<!--

.style1 {
```

```
font-size: 18px;
font-family: Georgia, "Times New Roman", Times, serif;
color: #FFFFFF;
}
.style3 {font-size: 18px}
.style4 {
color: #FFFFFF;
font-weight: bold;
}
.style5 {color: #FFFFFF}
-->
</style>
</head>
<body id="page1">
<div id="main">
<div class="extra-bg"></div>
<!-- header -->
<div id="header">
<div class="row-1">
<ul class="top-links">
<li><a href="#"><img alt="" src="images/home-icon.gif" /></a></li>
<li><a href="#"><img alt="" src="images/mail-icon.gif" /></a></li>
</ul>
```

```
<ul class="nav">

<li class="first"><a href="index.html">Home</a></li>

<li><a href="userlogin.jsp"><span>User</span></a> </li>

<li><a href="agentlogin.jsp">OSN</a></li>

</ul>

</div>

<div class="row-2 style6">

<p class="first style1 style3">Detecting Malicious Social Bots Based on
Clickstream Sequences</div>

</div>

<!-- content -->

<div id="content">

<div class="wrapper">

<div class="aside">

<div class="inner_copy">More <a href="#">Website Templates</a>
@ Templates.com!</div>

<div class="section">

<!-- box begin -->

<div class="box">

<div class="inner">

<h4><span><span>Menu</span></span></h4>

<div class="inner">

<ul>

<li><a href="index.html">Home</a></li>
```

User

OSN

<!-- box end -->

<!-- box1 begin -->

<!-- box1 end -->

<h2>Detecting Malicious Social Bots Based on Clickstream Sequences</h2>

With the signicant increase in the volume, velocity, and variety of user data (e.g., usergenerated data) in online social networks, there have been attempted to design new ways of collecting and analyzing such big data. For example, social bots have been used to perform automated analytical services and provide users with improved quality of service. However, malicious social bots have also been used to disseminate false information (e.g., fake news), and this can result in real-world consequences. Therefore, detecting and removing malicious social bots in online social networks is crucial. The most existing detection methods of malicious social bots analyze the quantitative features of their behavior. These features are easily imitated by social bots; thereby resulting in low accuracy of the analysis. A novel method of detecting malicious social bots, including both features selection based on the transition probability

of clickstream sequences and semi-supervised clustering, is presented in this paper. This method not only analyzes transition probability of user behavior clickstreams but also considers the time feature of behavior. Findings from our experiments on real online social network platforms demonstrate that the detection accuracy for different types of malicious social bots by the detection method of malicious social bots based on transition probability of user behavior clickstreams increases by an average of 12.8%, in comparison to the detection method based on quantitative analysis of user behavior.</p>

<!-- box2 begin -->

<h3 class="style5"> </h3>

<ul class="img-list style5">

<li class="last">

<!-- box2 end -->

```
</div>

<!-- footer -->

<div id="footer">

<ul class="nav">

<li></li></ul>

</div>

</div>

<script type="text/javascript"> Cufon.now(); </script>

<div align=center></div>

</body>

</html>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<title>users login page</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="style.css" rel="stylesheet" type="text/css" />

<script src="js/jquery-1.3.2.min.js" type="text/javascript"></script>

<script src="js/cufon-yui.js" type="text/javascript"></script>

<script src="js/cufon-replace.js" type="text/javascript"></script>

<script src="js/Myriad_Pro_400.font.js" type="text/javascript"></script>
```

```
<!--[if lt IE 7]>
<link href="ie_style.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="js/ie_png.js"></script>
<script type="text/javascript">ie_png.fix('.png, .extra-bg, .box, .box1, .box2, .img-
list img');</script>
<![endif]-->
<style type="text/css">
<!--
.style1 {
        font-size: 36px;
        font-family: Georgia, "Times New Roman", Times,
        serif; color: #FF3366;
}
.style2 {color: #999999}
.style3 {font-family: Georgia, "Times New Roman", Times, serif}
.style4 {color: #33FF00; }
.style6 {color: #0066FF}
.style7 {color: #FFFF00}
.style10 {color: #FFFFFF; font-weight: bold; }
.style15 {color: #0000FF; font-weight: bold; }
-->
```

```
</style>
</head>
<body id="page1">
<div id="main">
 <div class="extra-bg"></div>
 <!-- header -->
 <div id="header">
  <div class="row-1">
   <ul class="top-links">
    <li><a href="#"><img alt="" src="images/home-icon.gif" /></a></li>
    <li><a href="#"><img alt="" src="images/mail-icon.gif" /></a></li>
   </ul>
   <ul class="nav">
    <li class="first"><a href="index.html">Home</a></li>
        <li><a href="userlogin.jsp"><span>User</span></a> </li>
        <li><a href="register.jsp"><span>Register</span></a> </li>
            <li><a href="agentlogin.jsp">OSN</a></li>
   </ul>
  </div>
  <div class="row-2 style6">
   <p class="first  style1"> </p>
```

```
    </div>
  </div>
  <!-- content -->
  <div id="content">
   <div class="wrapper">
    <div class="aside">
     <div class="inner_copy">More <a href="#">Website Templates</a>
@ Templates.com!</div>
     <div class="section">
      <!-- box begin -->
      <div class="box">
       <div class="inner">
        <h4><span><span>Menu</span></span></h4>
        <div class="inner">
         <ul>
          <li><a href="index.html">Home</a></li>
                                <li><a href="userlogin.jsp">User</a></li>
                                <li><a href="agentlogin.jsp">OSN</a> </li>
                                </ul>
       </div>
      </div>
```

```
    </div>
    <!-- box end -->
   </div>
   <!-- box1 begin -->


   <!-- box1 end -->
  </div>
  <div class="mainContent">
   <div class="section">
    <h2 class="style3">User LoGIN </h2>
    <div class="indent">
     <form action="authentication.jsp?type=<%="user"%>" method="post">


    <p align="left"><img src="images/Login.jpg" alt="" width="140" height="90" /></p>
    <table width="480">
     <tr>
      <td width="162" height="48"><span class="style15">User
Name (required)</span></td>
          <td width="347"><input id="name" name="userid" class="text" /></td>
     </tr>
     <tr>
```

```
<td height="44"><span class="style15">Password (required)</span></td>
<td><input type="password" id="password" name="pass" class="text" /></td>
</tr>
<tr>
<td height="51"></td>
<td><span class="style2">
<input name="imageField" type="submit" class="style3" id="imageField" value="Login" />
<input name="Reset" type="reset" class="style3" value="Reset" />
<span class="style7"><span class="style10">New user</span>?</span></span><a href="register.jsp" class="style4">Register </a> </td>
</tr>
</table>
</form>
</div>
</div>
<!-- box2 begin -->
<div class="box2">
<div class="inner">
</div>
</li>
</ul>
```

```
        </div>
       </div>
       <!-- box2 end -->
      </div>
    </div>
  </div>
  <!-- footer -->
  <div id="footer">
   <ul class="nav">
    <li></li>
   </ul>
   <div class="wrapper aligncenter"></div>
  </div>
</div>
<script type="text/javascript"> Cufon.now(); </script>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
```

```
<head>
<title>users registration form</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="style.css" rel="stylesheet" type="text/css" />
<script src="js/jquery-1.3.2.min.js" type="text/javascript"></script>
<script src="js/cufon-yui.js" type="text/javascript"></script>
<script src="js/cufon-replace.js" type="text/javascript"></script>
<script src="js/Myriad_Pro_400.font.js" type="text/javascript"></script>
<!--[if lt IE 7]>
<link href="ie_style.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="js/ie_png.js"></script>
<script type="text/javascript">ie_png.fix('.png, .extra-bg, .box, .box1, .box2, .img-
list img');</script>
<![endif]-->
<style type="text/css">
<!--
.style1 {
        font-size: 36px;
        font-family: Georgia, "Times New Roman", Times,
        serif; color: #FF3366;
}
```

```
.style3 {font-family: Georgia, "Times New Roman", Times, serif}
.style4 {color: #99FF00}
.style6 {font-weight: bold}
.style7 {color: #FFFF00; font-weight: bold; }
-->
</style>
</head>
<body id="page1">
<div id="main">
 <div class="extra-bg"></div>
 <!-- header -->
 <div id="header">
  <div class="row-1">
   <ul class="top-links">
    <li><a href="#"><img alt="" src="images/home-icon.gif" /></a></li>
    <li><a href="#"><img alt="" src="images/mail-icon.gif" /></a></li>
   </ul>
   <ul class="nav">
    <li class="first"><a href="index.html">Home</a></li>
        <li><a href="userlogin.jsp"><span>User</span></a> </li>
        <li><a href="register.jsp"><span>Register</span></a> </li>
```

```
                <li><a href="agentlogin.jsp">OSN</a></li>
    </ul>
   </div>
   <div class="row-2 style6">
    <p class="first  style1"> </p>
    </div>
  </div>
  <!-- content -->
  <div id="content">
   <div class="wrapper">
    <div class="aside">
     <div class="inner_copy">More <a href="#">Website Templates</a>
@ Templates.com!</div>
     <div class="section">
      <!-- box begin -->
      <div class="box">
       <div class="inner">
        <h4><span><span>Menu</span></span></h4>
        <div class="inner">
         <ul>
          <li><a href="index.html">Home</a></li>
```

```
                                <li><a href="userlogin.jsp">User</a></li>
                                <li><a href="agentlogin.jsp">OSN</a> </li>
                                </ul>
      </div>
     </div>
   </div>
   <!-- box end -->
  </div>
  <!-- box1 begin -->

  <!-- box1 end -->
 </div>
 <div class="mainContent">
  <div class="section">
   <h2 class="style3">User ReGISTRATION Form </h2>
  </div>
  <!-- box2 begin -->
  <div class="box2">
   <div class="inner">
       <div align="justify">
                                <form action="insertdata.jsp" method="post" id=""
```

```
enctype="multipart/form-data">
                                        <p><img src="images/Register.png" alt=""
width="155" height="96" /></p>
                                        <table width="513">
                <tr>
                 <td height="43" bgcolor="#FF0000"><span
class="style7">Select Group(required) </span></td>
                 <td><select id="select" name="group" style="width:175px;" class="text">
                  <option>--Select--</option>
                  <option>Music</option>
                  <option>Sports</option>
                                                        <option>Education</option>
                                                        <option>Healthcare</option>
                                                        <option>Cinema</option>

                 </select></td>
                </tr>
                <tr>
                 <td width="181" height="43" bgcolor="#FF0000"><span
class="style7">User Name (required)</span></td>
                 <td width="320"><input id="name" name="userid" class="text" /></td>
                </tr>
```

```
<tr>
 <td height="43" bgcolor="#FF0000"><span class="style7">Password (required)</span></td>
 <td><input type="password" id="password" name="pass" class="text" /></td>
</tr>
<tr>
 <td height="43" bgcolor="#FF0000"><span class="style7">Email Address (required)</span></td>
 <td><input id="email" name="email" class="text" /></td>
</tr>
<tr>
 <td height="43" bgcolor="#FF0000"><span class="style7">Mobile Number (required)</span></td>
 <td><input id="mobile" name="mobile" class="text" /></td>
</tr>
<tr>
 <td height="43" bgcolor="#FF0000"><span class="style7">Date of Birth (required)</span></td>
 <td><input id="dob" name="dob" class="text" /></td>
</tr>
<tr>
 <td height="43" bgcolor="#FF0000"><span class="style7">Select Gender
```

```
(required)</span></td>
                <td><select id="s1" name="gender" style="width:175px;" class="text">
                 <option>--Select--</option>
                 <option>MALE</option>
                 <option>FEMALE</option>
                </select></td>
               </tr>
               <tr>
                <td height="65"
bgcolor="#FF0000"><span
class="style7">Address</span></td>
                <td><textarea id="address" name="address" rows="3"
cols="25"></textarea></td>
               </tr>
               <tr>
                <td height="43" bgcolor="#FF0000"><span class="style7">Enter
Pincode (required)</span></td>
                <td><input id="pincode" name="pincode" class="text" /></td>
               </tr>
               <tr>
                <td height="43" bgcolor="#FF0000"><span class="style7">Select Profile
Picture(required) </span></td>
                <td><input type="file" id="pic" name="pic" class="text" /></td>
```

```
            </tr>
            <tr>
              <td height="43" rowspan="3">
                <p> </p></td>
              <td align="left" valign="middle"> <p> 
                </p>
                <p>
                  <input name="submit" type="submit" value="REGISTER" />
                </p>
                <div align="right">
                <p align="right"><a href="userlogin.jsp" class="style4">Back</a></p>
              </div></td>
            </tr>
        </table>
      </form>


                              </div>
    </div>
  </div>
  <!-- box2 end -->
</div>
```

```
  </div>
 </div>
 <!-- footer -->
 <div id="footer">
  <ul class="nav">
   <li></li>
  </ul>
  <div class="wrapper aligncenter"></div>
 </div>
</div>
<script type="text/javascript"> Cufon.now(); </script>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<title>OSN Results</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="style.css" rel="stylesheet" type="text/css" />
```

```
<script src="js/jquery-1.3.2.min.js" type="text/javascript"></script>
<script src="js/cufon-yui.js" type="text/javascript"></script>
<script src="js/cufon-replace.js" type="text/javascript"></script>
<script src="js/Myriad_Pro_400.font.js" type="text/javascript"></script>
<!--[if lt IE 7]>
<link href="ie_style.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="js/ie_png.js"></script>
<script type="text/javascript">ie_png.fix('.png, .extra-bg, .box, .box1, .box2, .img-
list img');</script>
<![endif]-->
<style type="text/css">
<!--
.style1 {
        font-size: 36px;
        font-family: Georgia, "Times New Roman", Times,
        serif; color: #FF3366;
}
.style2 {color: #999999}
.style3 {font-family: Georgia, "Times New Roman", Times, serif}
.style4 {color: #33FF00; }
.style6 {color: #0066FF}
```

```
.style7 {color: #FFFF00}
-->
</style>
</head>
<body id="page1">
<div id="main">
 <div class="extra-bg"></div>
 <!-- header -->
 <div id="header">
  <div class="row-1">
   <ul class="top-links">
    <li><a href="#"><img alt="" src="images/home-icon.gif" /></a></li>
    <li><a href="#"><img alt="" src="images/mail-icon.gif" /></a></li>
   </ul>
        <%String agent=(String)application.getAttribute("agent");%>
   <ul class="nav">
    <li class="first"><a href="agentmain.jsp">Home</a></li>
        <li><a href="#"><span><%=agent%></span></a> </li>
            <li><a href="agentlogin.jsp">Logout</a></li>
   </ul>
  </div>
```

```
 <div class="row-2 style6">
  <p class="first  style1"> </p>
  </div>
</div>
<!-- content -->
<div id="content">
 <div class="wrapper">
  <div class="aside">
   <div class="inner_copy"></div>
   <div class="section">
    <!-- box begin -->
    <div class="box">
     <div class="inner">
      <h4><span><span>Menu</span></span></h4>
      <div class="inner">
       <ul>
        <li>
         <h6><a href="usermain.jsp">Home</a></h6>
        </li>
                                <li></li>
                                <li>
```

```
                                    <h6><a href="agentlogin.jsp">Logout</a> </h6>
                                   </li>
                                </ul>
                    </div>
                   </div>
                  </div>
                  <!-- box end -->
                 </div>
                 <!-- box1 begin -->

                 <!-- box1 end -->
                </div>
                <div class="mainContent">
                 <div class="section">
                  <h2 class="style3">OSN Results </h2>
                  <div class="indent">


                    <iframe width="555" height="450"
src="results.jsp?uname=<%=request.getParameter("uname") %>"></iframe>
                    <p> </p>
```

```
        <a href="agentmain.jsp">Back</a></div>
      </div>
      <!-- box2 begin -->
      <div class="box2">
        <div class="inner"></div>
      </div>
      <!-- box2 end -->
     </div>
    </div>
   </div>
   <!-- footer -->
   <div id="footer">
    <ul class="nav">
     <li></li>
    </ul>
    <div class="wrapper aligncenter"></div>
   </div>
  </div>
  <script type="text/javascript"> Cufon.now(); </script>
  <div align=center></div>
  </body></html>
```

7. SYSTEM TESTING

7. SYSTEM TESTING

Software once validated must be combined with other system elements (e.g. Hardware, people and database). System testing verifies that all the elements are proper and that overall system function performance is achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

7.1 TESTING METHODOLOGIES

The following are the Testing Methodologies:

- Unit Testing.
- Integration Testing.
- User Acceptance Testing.
- Output Testing.
- Validation Testing.

Unit Testing:

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module's control structure to ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.

During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All important processing path are tested for the expected results. All error handling paths are also tested.

Integration Testing:

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

The following are the types of Integration Testing:

1. Top Down Integration:

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

2. Bottom-up Integration:

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

3. User Acceptance Testing:

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

4. Output Testing:

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

Validation Checking:

Validation checks are performed on the following fields.

Text Field:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Preparation of Test Data:

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

Using Live Test Data:

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

Using Artificial Test Data:

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

The package “Virtual Private Network” has satisfied all the requirements specified as per software requirement specification and was accepted.

USER TRAINING:

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

7.2 TESTING STRATEGY:

A strategy for system testing integrates system test cases and design techniques into a well-planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

White Box Testing:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

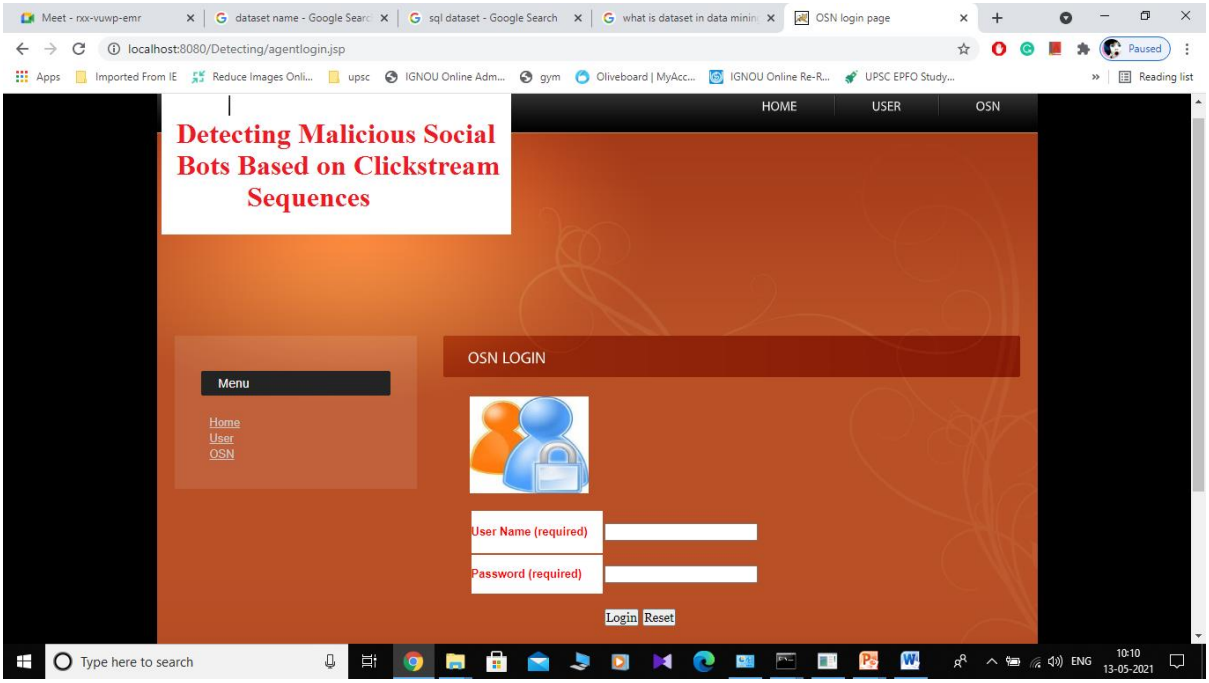
7.3 MAINTAINENCE

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user’s requirements during the process of system development With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing is simple and easy to understand which will make maintenance easier.

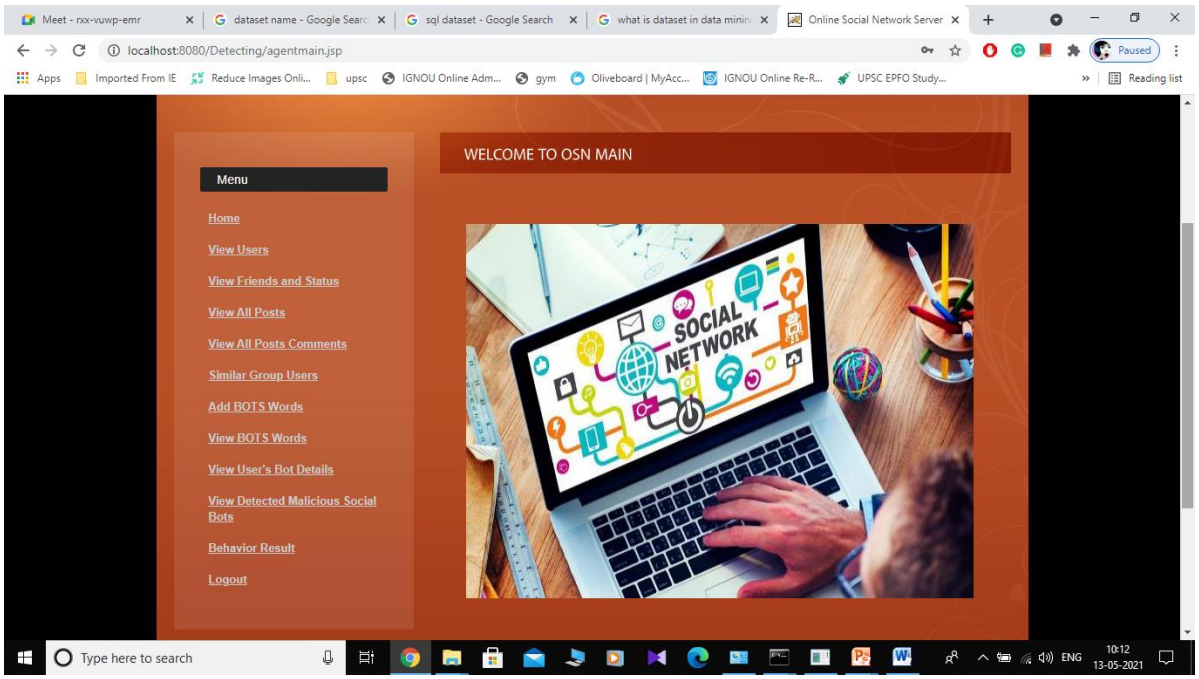
8. OUTPUT SCREENS

8. OUTPUT SCREENS:

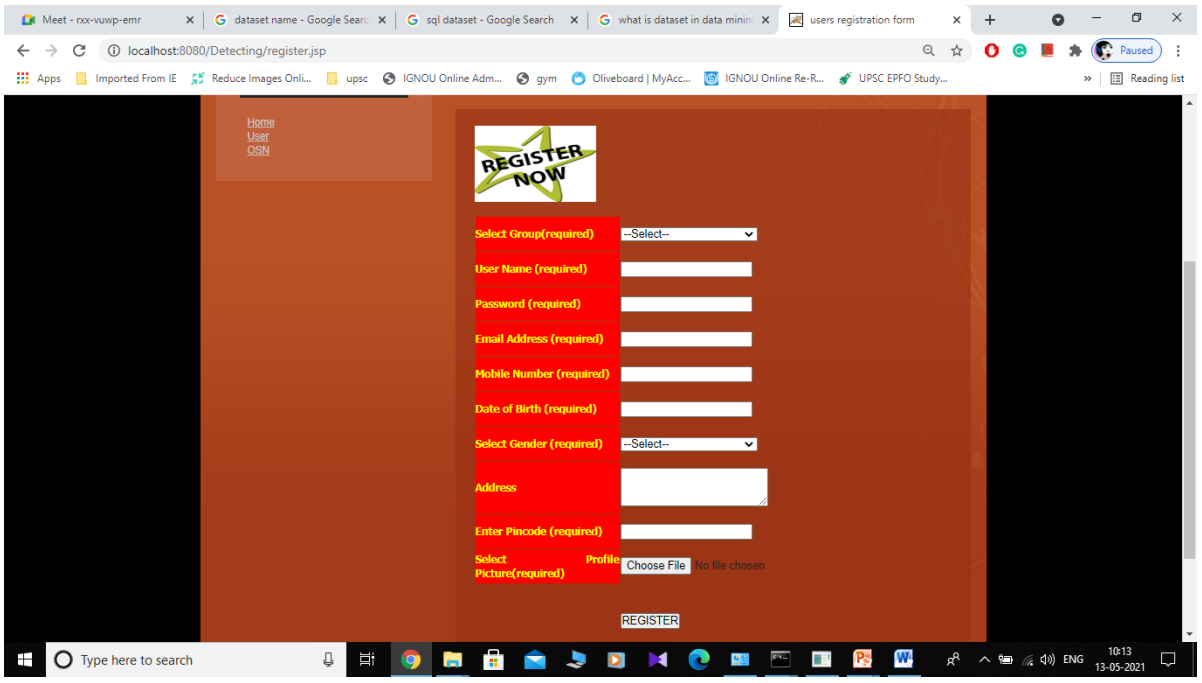
8.1 : HOME SCREEN OF OSN



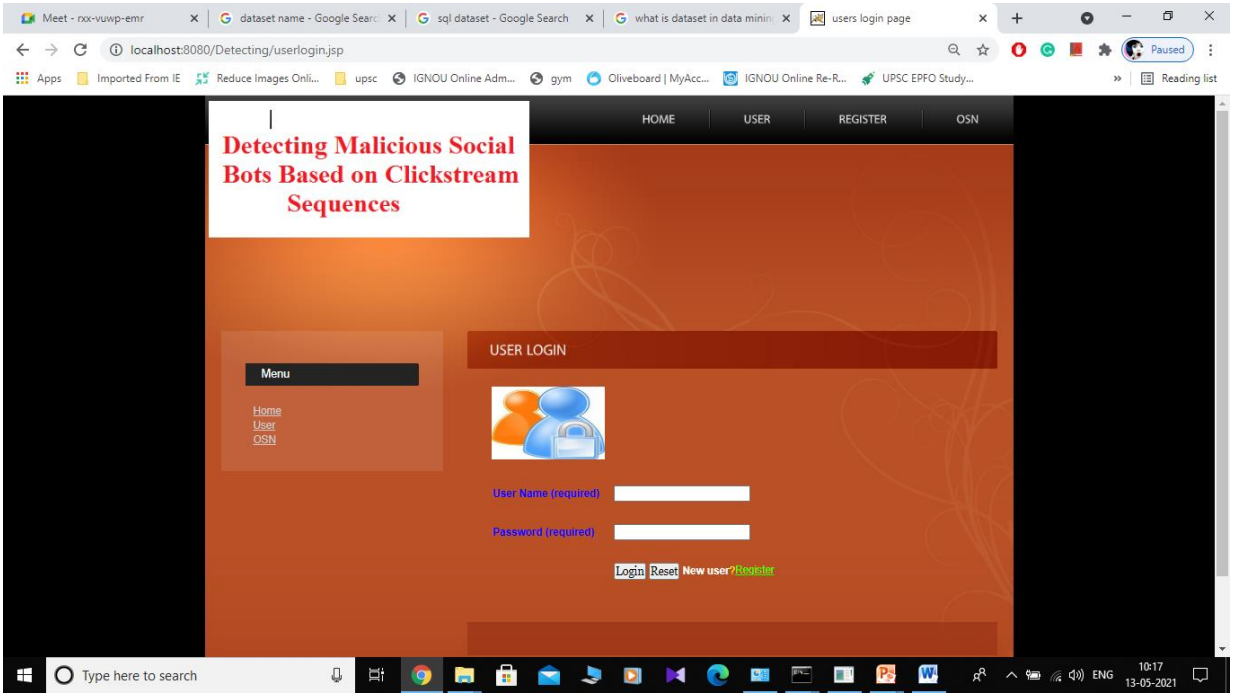
8.2 : OSN LOGIN PAGE



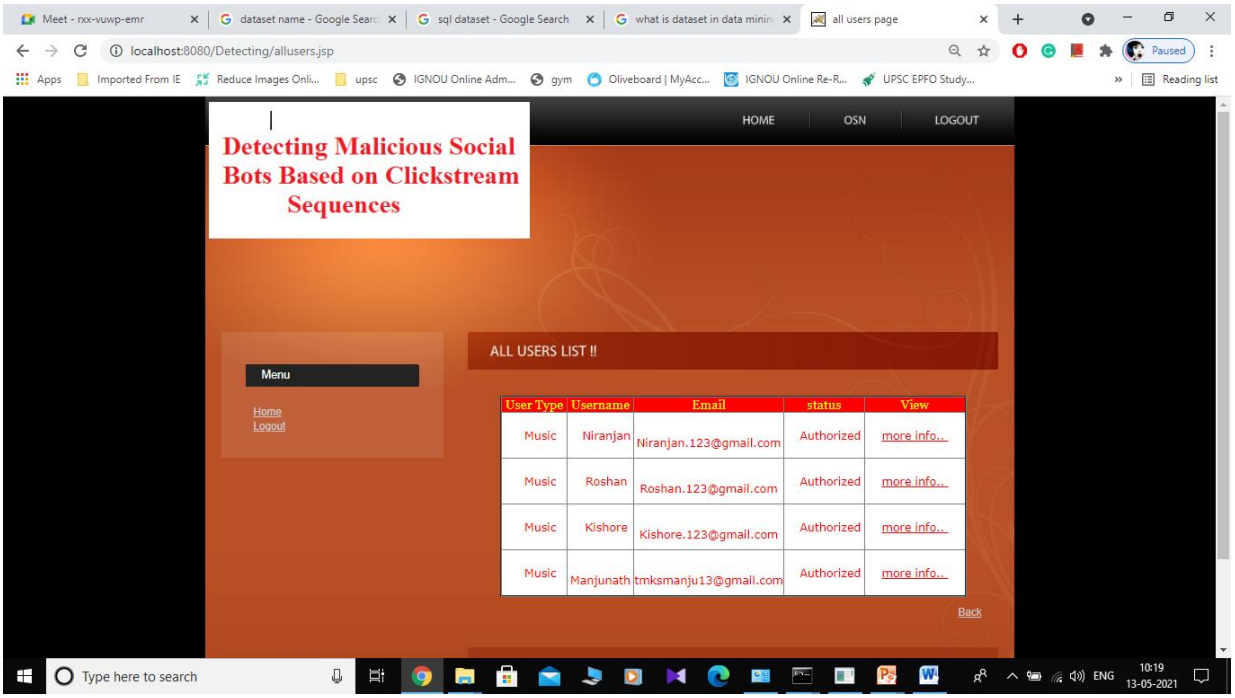
8.3 : OSN MAIN PAGE



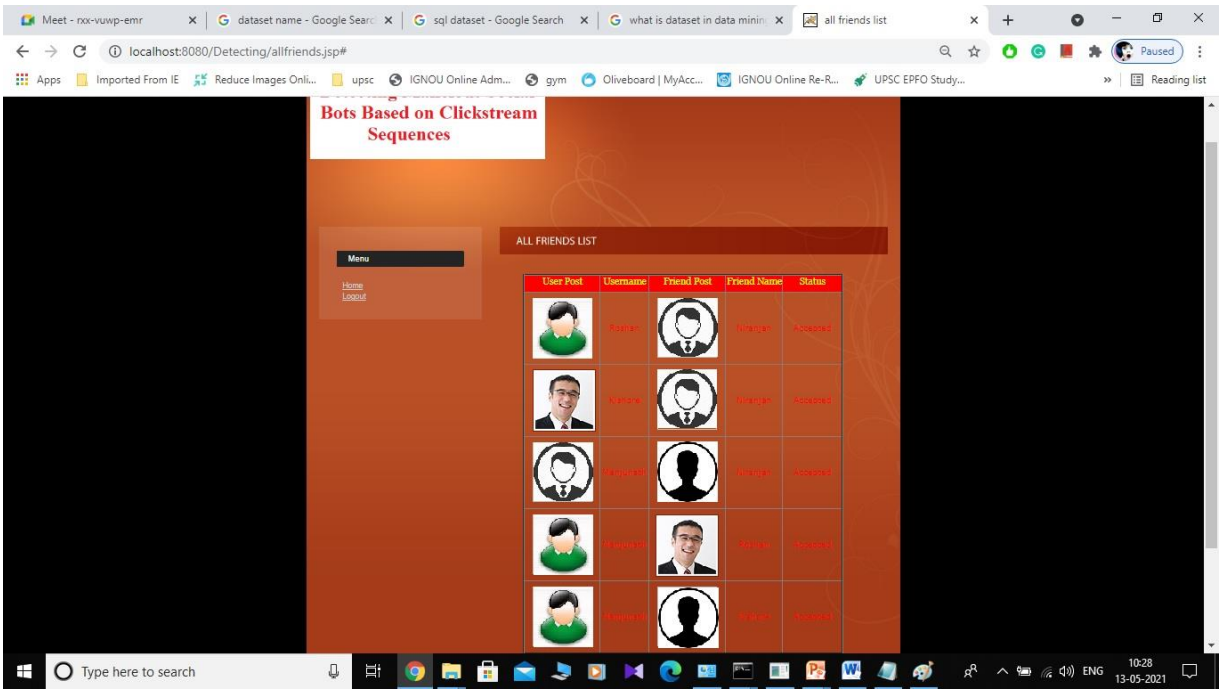
8.4 : USER REGISTRATION FORM



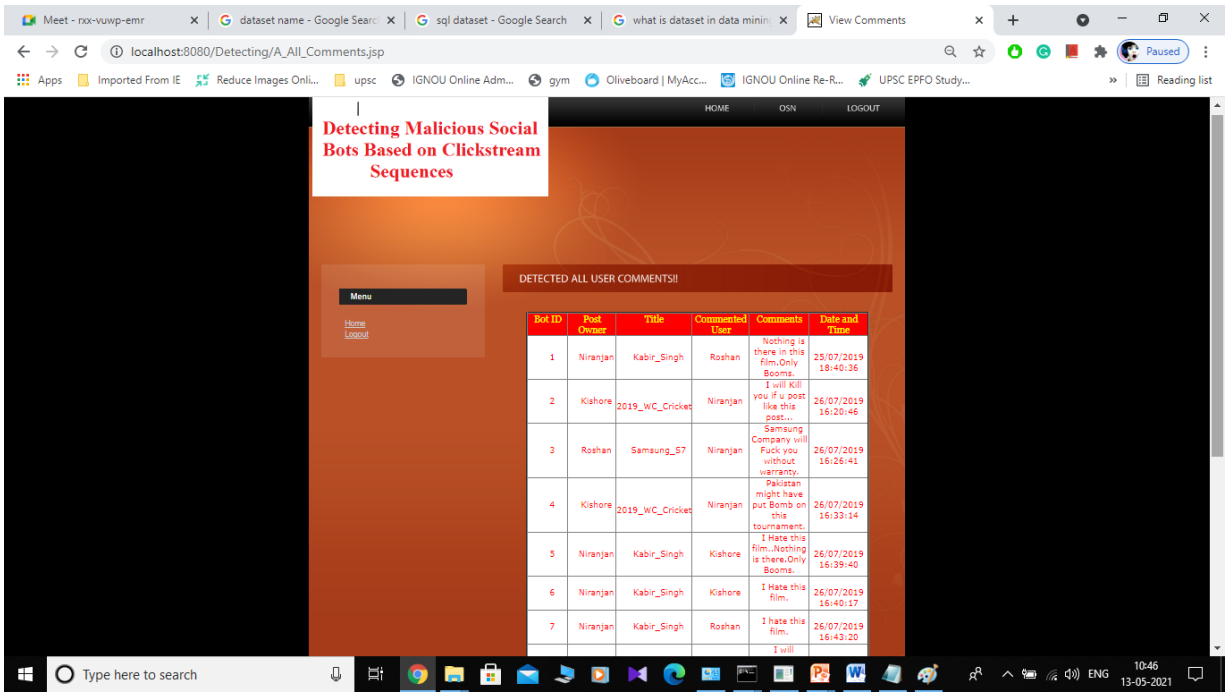
8.5 : USER LOGIN PAGE



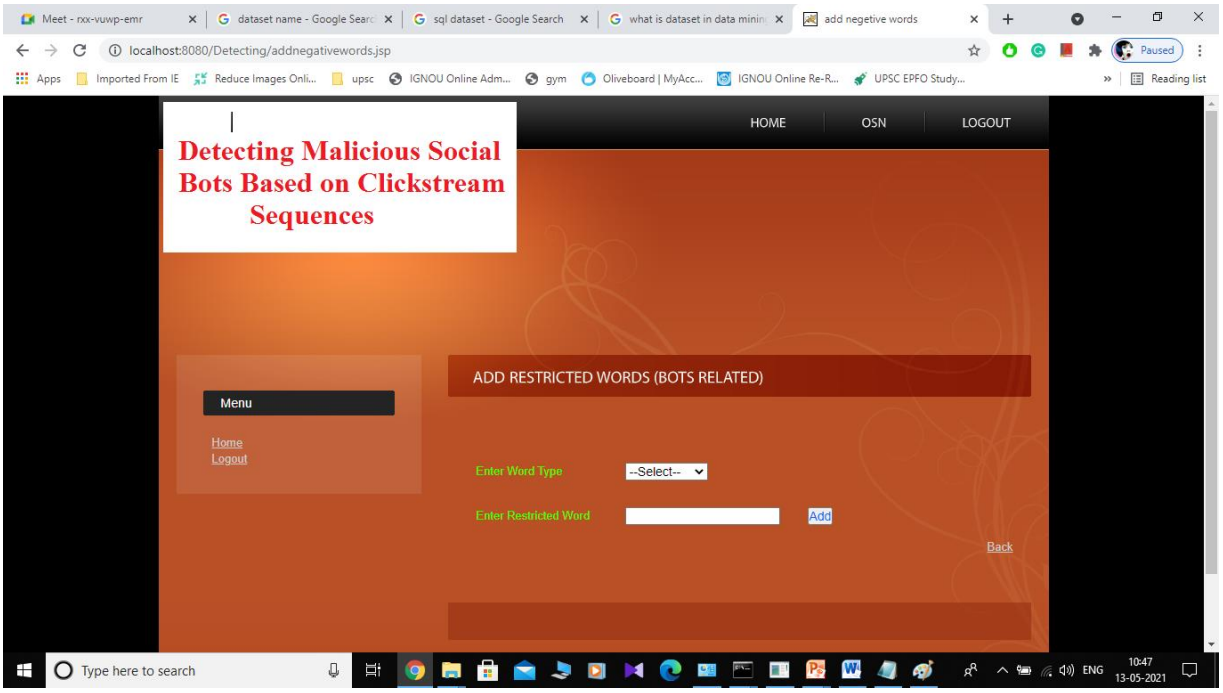
8.6 : ALL USERS APGE



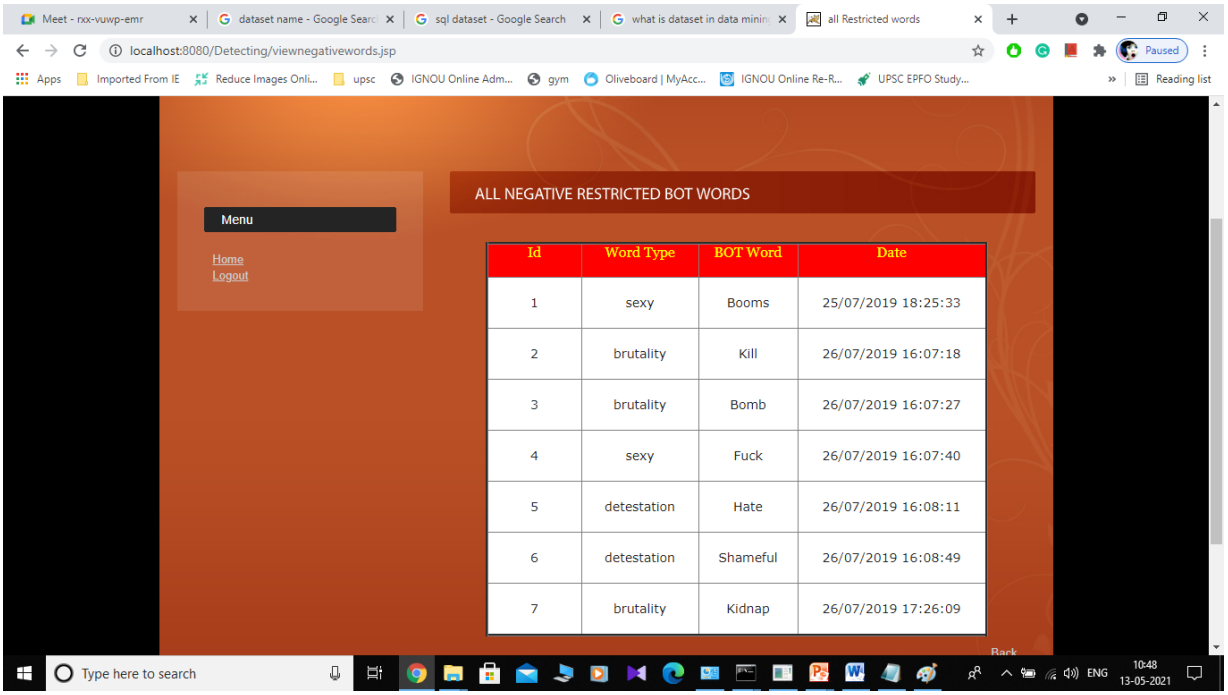
8.7 : ALL FRIENDS LIST



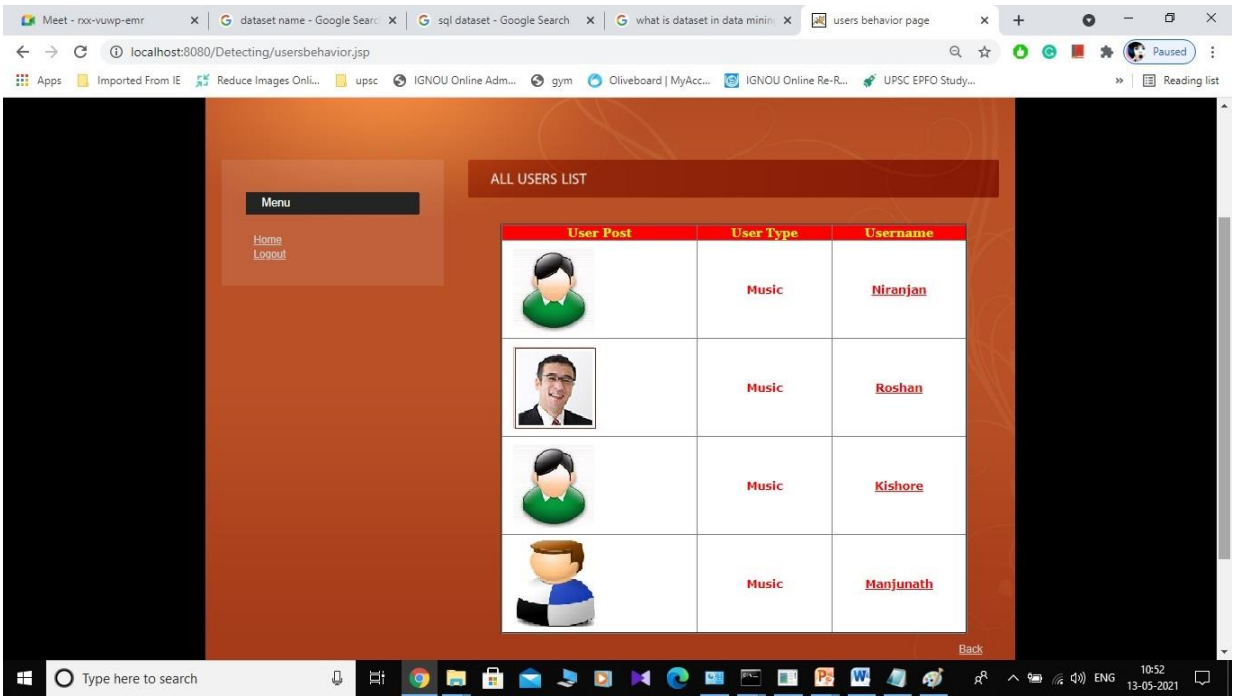
8.8 : COMMENTS VIEW



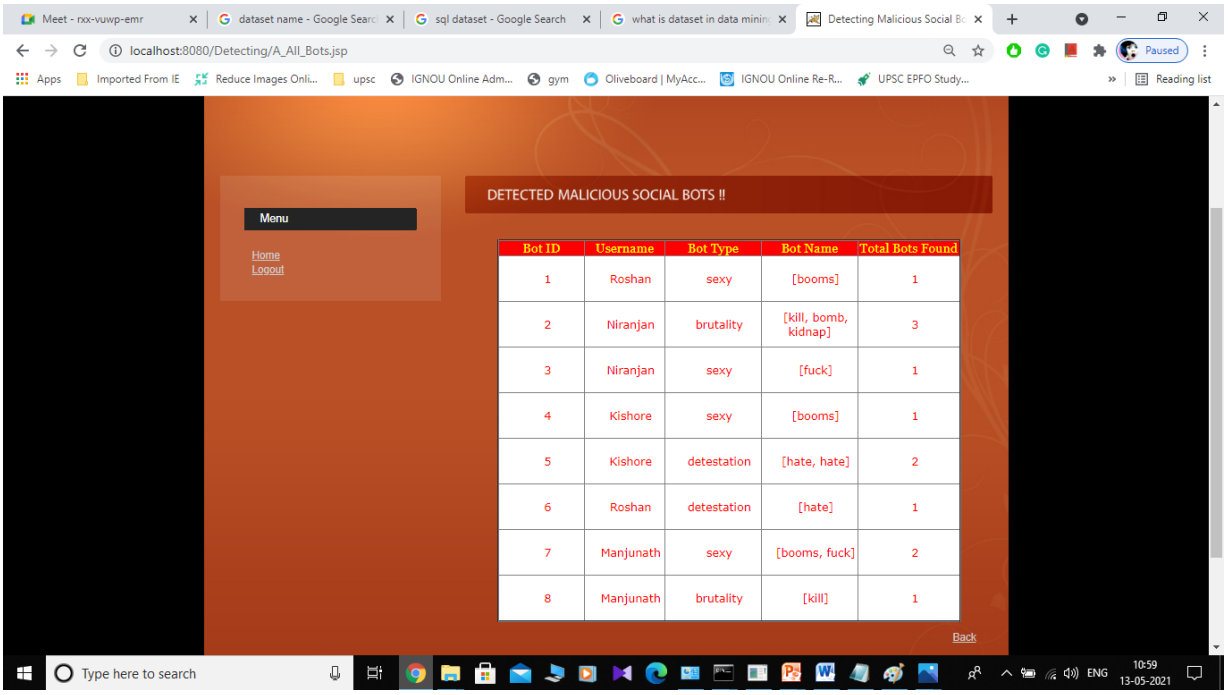
8.9 : ADD NEGATIVE WORDS



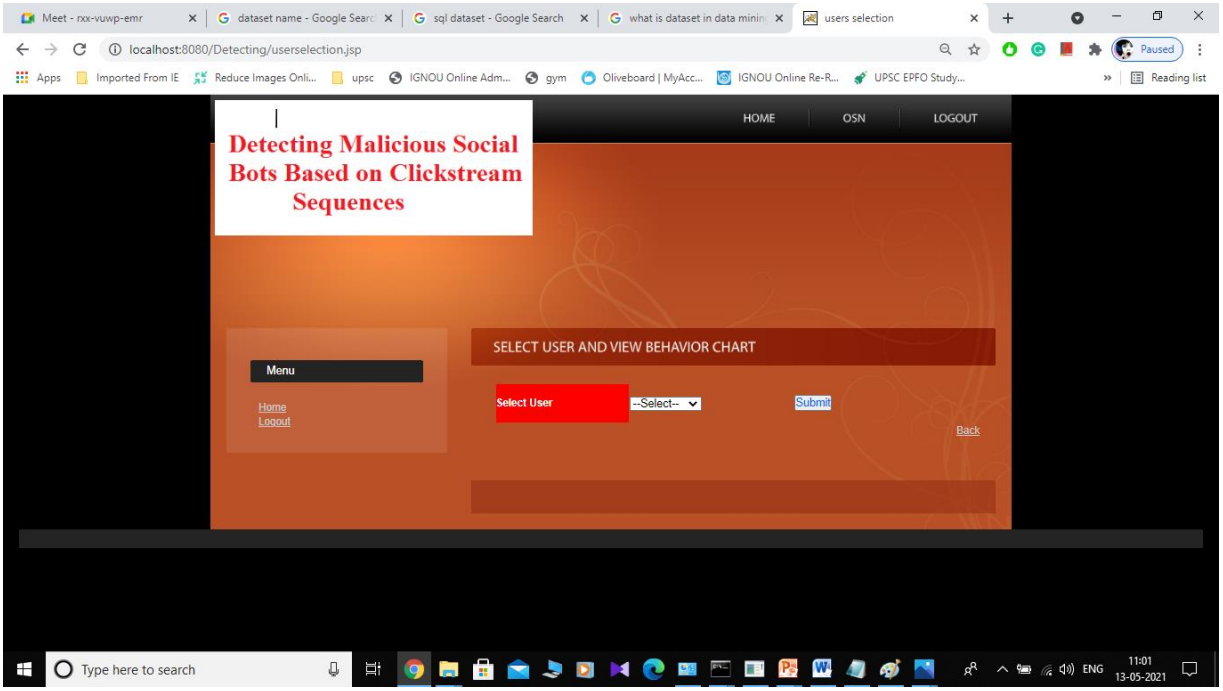
8.10 : ALL RESTRICTED WORDS



8.11 : USERS BEHAVIOUR PAGE

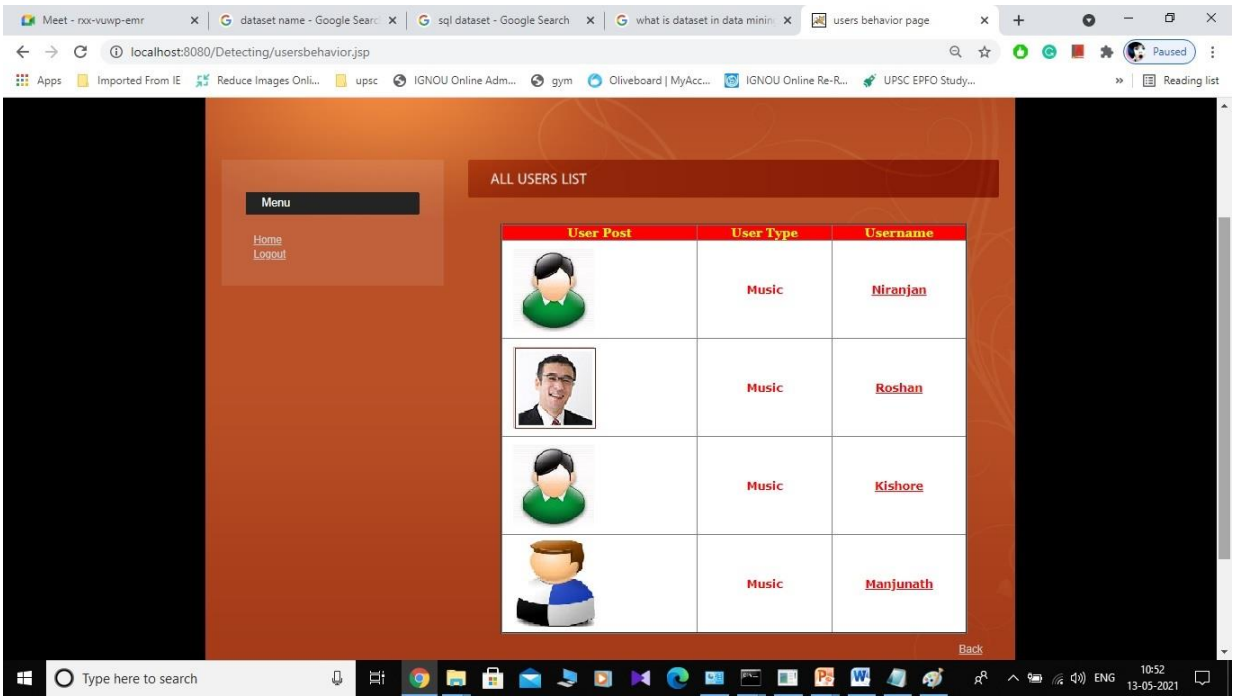


8.12 : BOT DETECTION

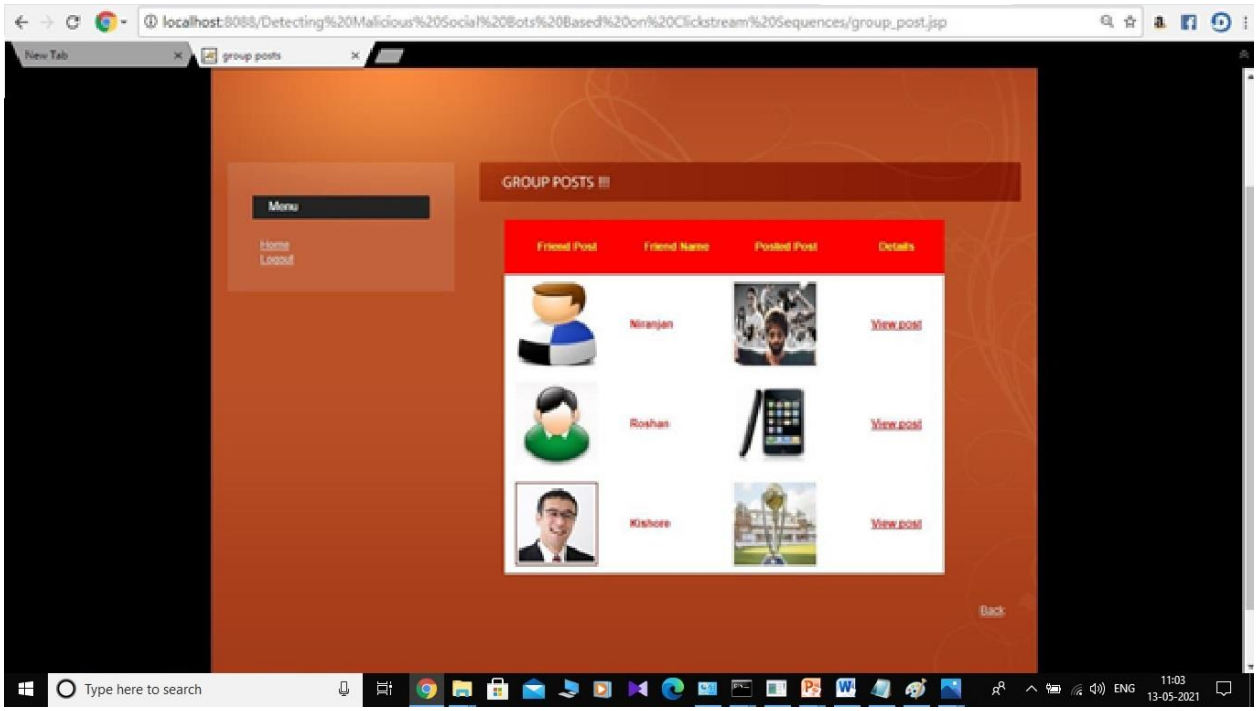


8.13 : USERS SELECTION

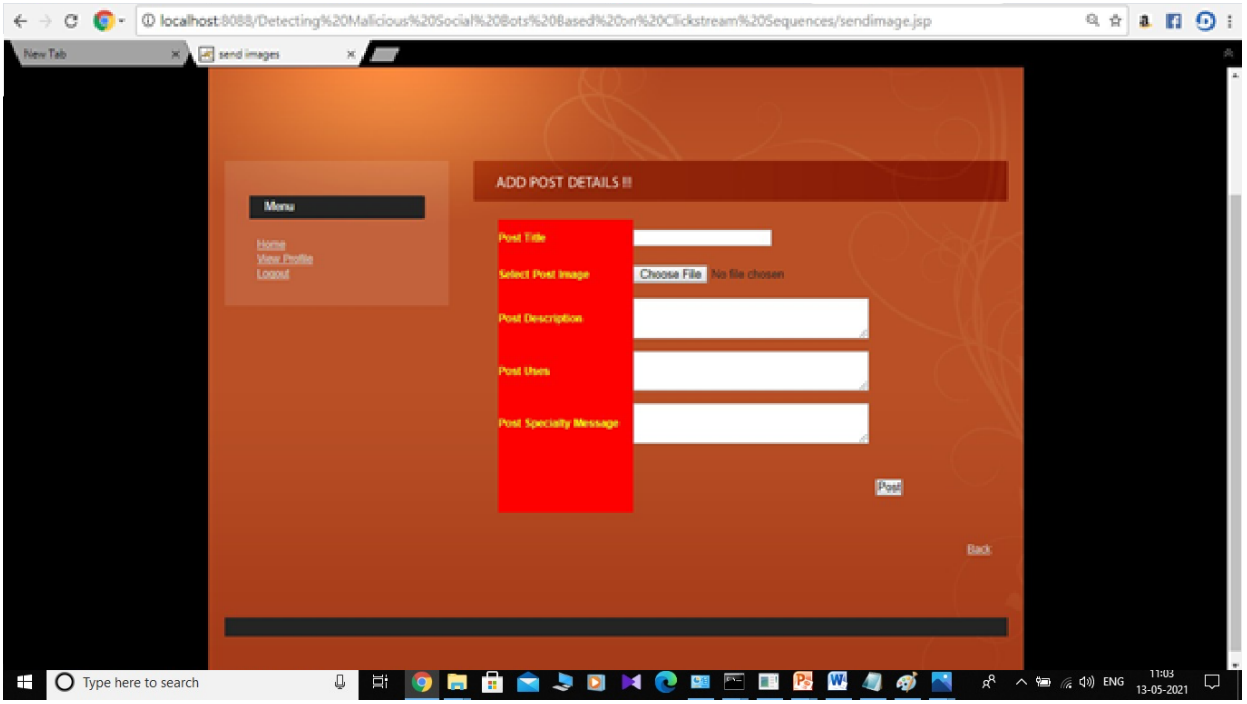
8.14 : USER DETAILS



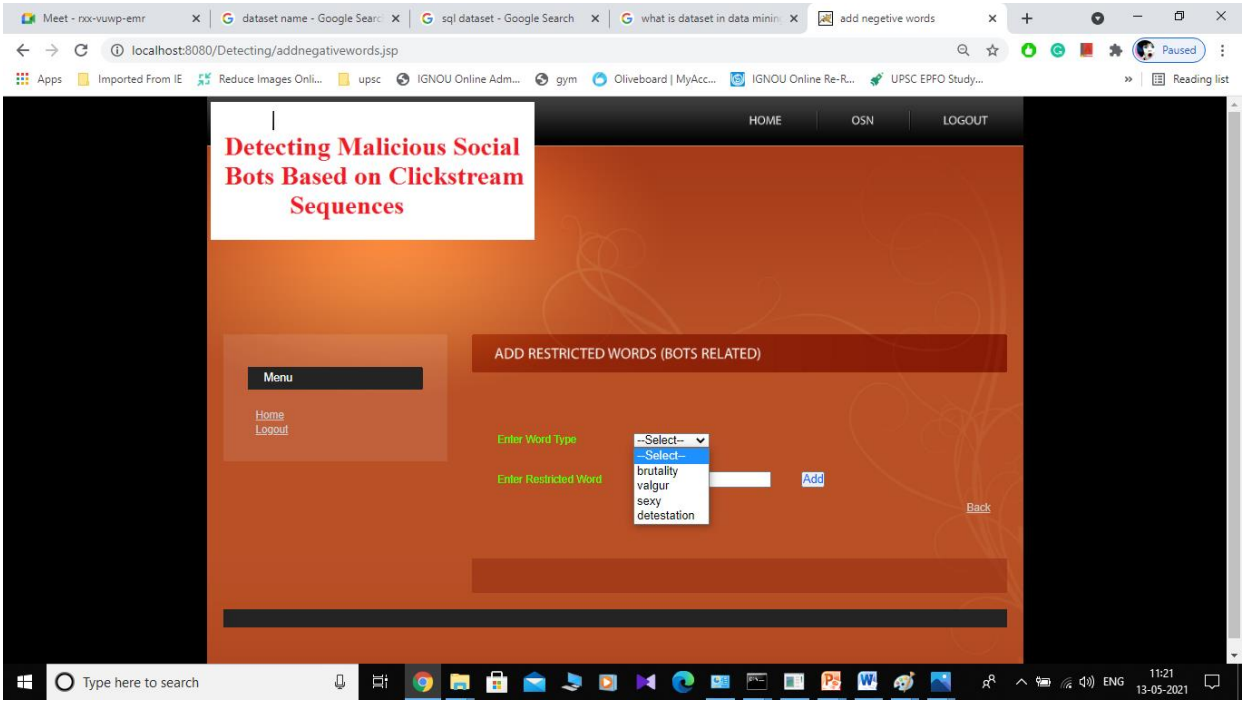
8.15 : DELETE FRIENDS



8.16 : GROUP POSTS



8.17 : SEND IMAGES



8.18 : ADD NEGATIVE WORDS

9. CONCLUSION

9. CONCLUSION

We proposed a novel method to accurately detect malicious social bots in online social networks. Experiments showed that transition probability between user click streams based on the social situation analytics can be used to detect malicious social bots in online social platforms accurately. In future research, additional behaviors of malicious social bots will be further considered and the proposed detection approach will be extended and optimized to identify specific intentions and purposes of a broader range of malicious social bots.

10.BIBLIOGRAPHY AND REFERENCES

10. BIBLIOGRAPHY AND REFERENCES

Oracle

PL/SQL Programming by Scott Urman

SQL complete reference by Livion

JAVA Technologies

JAVA Complete Reference

Java Script Programming by Yehuda Shiran

Mastering JAVA Security

JAVA2 Networking by Pistoria

JAVA Security by Scotl oaks

Head First EJB Sierra Bates

J2EE Professional by Shadab siddiqui

JAVA server pages by Larne Pekowsley

JAVA Server pages by Nick Todd

HTML

HTML Black Book by Holzner

JDBC

Java Database Programming with JDBC by Patel moss.

Software Engineering by Roger Pressman

REFERENCES:

[1] F. Morstatter, L. Wu, T. H. Nazer, K. M. Carley, and H. Liu, ``A new approach to bot detection: Striking the balance between precision and recall," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, San Francisco, CA, USA, Aug. 2016, pp. 533_540.

[2] C. A. De Lima Salge and N. Berente, ``Is that social bot behaving unethically?" *Commun. ACM*, vol. 60, no. 9, pp. 29_31, Sep. 2017.

[3] M. Sahlabadi, R. C. Muniyandi, and Z. Shukur, ``Detecting abnormal Behavior in social network Websites by using a process mining technique,"*J. Comput. Sci.*, vol. 10, no. 3, pp. 393_402, 2014.

[4] F. Brito, I. Petiz, P. Salvador, A. Nogueira, and E. Rocha, ``Detectingsocial-network bots based on multiscale behavioral analysis," in *Proc.7th Int. Conf. Emerg. Secur. Inf., Syst. Technol. (SECURWARE)*, Barcelona,Spain, 2013, pp. 81_85.

[5] T.-K. Huang, M. S. Rahman, H. V. Madhyastha, M. Faloutsos, andB. Ribeiro, ``An analysis of socware cascades in online social networks,"in *Proc. 22nd Int. Conf. World Wide Web*, Rio de Janeiro, Brazil, 2013,pp. 619_630.

[6] H. Gao *et al.*, ``Spam ain't as diverse as it seems: Throttling OSN spam withtemplates underneath," in *Proc. 30th ACSAC*, New Orleans, LA, USA,2014, pp. 76_85.

[7] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, ``The rise ofsocial bots," *Commun. ACM*, vol. 59, no. 7, pp. 96_104, Jul. 2016.

[8] T. Hwang, I. Pearce, and M. Nanis, ``Socialbots: Voices from the fronts,"*Interactions*, vol. 19, no. 2, pp. 38_45, Mar. 2012.

[9] Y. Zhou *et al.*, ``*ProGuard*: Detecting malicious accounts in socialnetwork-based online promotions," *IEEE Access*, vol. 5, pp. 1990_1999,2017.

[10] Z. Zhang, C. Li, B. B. Gupta, and D. Niu, ``Ef_cient compressedciphertext length scheme using multi-authority CP-ABE for hierarchicalattributes," *IEEE Access*, vol. 6, pp. 38273_38284, 2018. doi:10.1109/ACCESS.2018.2854600.

[11] C. Cai, L. Li, and D. Zengi, ``Behavior enhanced deep bot detection insocial media," in *Proc. IEEE Int. Conf. Intell. Secur. Inform. (ISI)*, Beijing,China, Jul. 2017, pp. 128_130.

[12] C. K. Chang, ``Situation analytics: A foundation for a new software engineering paradigm," *Computer*, vol. 49, no. 1, pp. 24_33, Jan. 2016.

[13] Z. Zhang, R. Sun, X. Wang, and C. Zhao, ``A situational analytic method for user behavior pattern in multimedia social networks," *IEEE Trans. BigData*, to be published. doi: 10.1109/TBDATA.2017.2657623.

[14] S. Barbon, Jr., G. F. C. Campos, G. M. Tavares, R. A. Igawa, M. L. Proença, Jr., and R. C. Guido, ``Detection of human, legitimate bot, and malicious bot in online social networks based on wavelets," *ACM Trans. MultimediaComput., Commun., Appl.*, vol. 14, no. 1s, Feb. 2018, Art. no. 26.

[15] J. Y. Park, N. O'Hare, R. Schifanella, A. Jaimes, and C.-W. Chung,``A large-scale study of user image search behavior on the Web," in *Proc.33rd Annu. ACM Conf. Hum. Factors Comput. Syst.*, Seoul, South Korea, 2015, pp. 985_994.

[16] G. Wang, X. Zhang, S. Tang, C. Wilson, H. Zheng, and B. Y. Zhao, ``Clickstream user behavior models," *ACM Trans. Web*, vol. 11, no. 4, Jul. 2017, Art. no. 21.

[17] Y. Liu, C.Wang, M. Zhang, and S. Ma, ``User behavior modeling for better Web search ranking," *Front. Comput. Sci.*, vol. 11, no. 6, pp. 923_936,Dec. 2017.

[18] M. Al-Qurishi, M. S. Hossain, M. Alrubaian, S. M. M. Rahman, andA. Alamri, ``Leveraging analysis of user behavior to identify maliciousactivities in large-scale social networks," *IEEE Trans. Ind. Informat.*,vol. 14, no. 2, pp. 799_813, Feb. 2018.

[19] J. Liu, Y. Xiao, K. Ghaboosi, H. Deng, and J. Zhang, ``Botnet: Classi_-cation, attacks, detection, tracing, and preventive measures," *EURASIPJ. Wireless Commun. Netw.*, vol. 2009, Dec. 2009, Art. no. 692654. doi:10.1155/2009/692654.

[20] Z. Chu, S. Gianvecchio, H.Wang, and S. Jajodia, ``Detecting automation oftwitter accounts: Are you a human, bot, or cyborg?" *IEEE Trans. Depend.Sec. Comput.*, vol. 9, no. 6, pp. 811_824, Nov. 2012.

[21] E. Van Der Walt and J. Eloff, ``Using machine learning to detectfake identities: Bots vs humans," *IEEE Access*, vol. 6, pp. 6540_6549,Jan. 2018.

[22] C. A. Davis, O. Varol, E. Ferrara, A. Flammini, and F. Menczer,``BotOrNot: A system to evaluate social bots," in *Proc. 25th Int. Conf.Companion World Wide Web*, Montreal, Canada, 2016, pp. 273_274.

[23] M. Fazil and M. Abulaish, ``Identifying active, reactive, and inactive targetsof socialbots in Twitter," in *Proc. Int. Conf. Web Intell.*, Leipzig, Germany,2017, pp. 573_580.

[24] A. F. Costa, Y. Yamaguchi, A. J. M. Traina, C. Traina, Jr., and C. Faloutsos,``Modeling temporal activity to detect anomalous behavior in socialmedia," *ACM Trans. Knowl. Discovery Data*, vol. 11, no. 4, Aug. 2017,Art. no. 49.

[25] S. Basu, A. Banerjee, and R. Mooney, ``Semi-supervised clustering byseeding," in *Proc. 19th Int. Conf. Mach. Learn.*, Sydney, NSW, Australia,2002, pp. 19_26.

[26] Z. Zhang, R. Sun, C. Zhao, J. Wang, C. K. Chang, and B. B. Gupta,``CyVOD:Anovel trinity multimedia social network scheme," *MultimediaTools Appl.*, vol. 76, no. 18, pp. 18513_18529, Sep. 2017.

DETECTING AND REMOVING MALICIOUS SOCIAL BOTS

Dr. M. VARAPRASAD RAO
Professor
Department of CSE
CMR TECHNICAL CAMPUS, HYDERABAD

A. Sahana Reddy
B.Tech student
Department of IT
CMR TECHNICAL CAMPUS, HYDERABAD

B. Devika
B.Tech student
Department of IT
CMR TECHNICAL CAMPUS,HYDERABAD

T. Vijayalaxmi
B.Tech student
Department of IT
CMR TECHNICAL CAMPUS,HYDERABAD

ABSTRACT

With the significant increase in the volume, velocity, and variety of user data (e.g., user generated data) in online social networks, there have been attempted to design new ways of collecting and analyzing such big data. For example, social bots have been used to perform automated analytical services and provide users with improved quality of service. However, malicious social bots have also been used to disseminate false information (e.g., fake news), and this can result in real-world consequences. Therefore, detecting and removing malicious social bots in online social networks is crucial. The most existing detection methods of malicious social bots analyze the quantitative features of their behavior. These features are easily imitated by social bots; thereby resulting in low accuracy of the analysis. A novel method of detecting malicious social bots, including both features selection based on the transition probability of clickstream sequences and semi-supervised clustering, is presented in this paper. This method not only analyzes transition probability of user behavior clickstreams but also considers the time feature of behavior. Findings from our experiments on real online social network platforms demonstrate that the detection accuracy for different types of malicious social bots by the detection method of malicious social bots based on transition probability of user behavior clickstreams increases by an average of 12.8%, in comparison to the detection method based on quantitative analysis of userbehavior.

INTRODUCTION

In online social networks, social bots are social accounts controlled by automated programs that can perform corresponding operations based on a set of procedures. The increasing use of mobile devices (e.g., Android and iOS devices) also contributed to an increase in the frequency and nature of user interaction via social networks. It is evidenced by the significant volume, velocity and variety of data generated from the large online social network

user base. Social bots have been widely deployed to enhance the quality and efficiency of collecting and analyzing data from social network services. However, public opinion about social networks and massive user data can also be mined or disseminated for malicious or nefarious purpose. In online social networks, automatic social bots cannot represent the real desires and intentions of normal human beings, so they are usually looked upon malicious one. In previous research, various methods were used to protect the security of online social network. User behavior is the most direct manifestation of user intent, as different users have different habits, preferences, and online behavior. Detect malicious social bots, and reduce the harm of malicious social bots, we need to acquire and analyze social situation of user behavior and compare and understand the differences of malicious social bots and normal users in dynamic behavior.

LITERATURESURVEY

H. Gao*et al.*, ``Spam ain't as diverse as it seems: Throttling OSN spam with templates underneath' For example, some fake social bots accounts created to imitate the profile of a normal user, steal user data and compromise their privacy [4], disseminate malicious or fake information [5], [6], malicious comment, promote or advance certain political or ideology agenda and propaganda [7], and influence the stock market and other societal and economical markets [8]

M. Al–Qurishi, M. S. Hossain, M. Alrubaian, S. M. M. Rahman, and A. Alamri, ``Leveraging analysis of user behavior to identify malicious activities in large–scale social networks.collected a large amount of user information on the Twitter and YouTube, about 13 million channel activities, analyzing and detecting abnormal behaviors that deviate significantly from large–scale specifications through user behavior in two social networks

E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, ``The rise of social bots,"For example, some fake social bots accounts created to imitate the profile of a normal user, steal user data and compromise their privacy , disseminate malicious or fake information,malicious comment, promote or advance certain political or ideology agenda and propaganda, and influence the stock market and other societal and economical markets .

F. Morstatter, L. Wu, T. H. Nazer, K. M. Carley, and H. Liu, ``A new approach to bot detection: Striking the balance between precision and recall,". Proposed a heuristic–type supervised BoostOR model with increasing recall rate to detect malicious bots, which using the proportion of tweets forwarded to the published tweets on the Twitter, the mean length of tweets, URL, and forwarding interval.

EXISTING SYSTEMS

The most existing detection methods of malicious social bots analyze the quantitative features of their behavior. These features are easily imitated by social bots; thereby resulting in low accuracy of the analysis. A novel method of detecting malicious social bots, including both features selection based on the transition probability of clickstream sequences and semi–supervised clustering, is presented.

PROPOSED SYSTEM

Proposed that situation analytics can be included in software service requirement analysis, which can facilitate the analysis of any change in user's requirements. Such an analysis is useful to understand the dynamic needs of a software service environment.

Proposed a heuristic–type supervised Boost OR model with increasing recall rate to detect malicious bots, which using the proportion of tweets forwarded to the published tweets on the Twitter, the mean length of tweets, URL, and forwarding interval.According to the social interactions between users of the Twitter user to identify the active, passive and inactive users, a supervised machine learning method was proposed to identify social bots on the basis of age, location and other static features of active, passive, and inactive users in the Twitter, as well as interacting person, interaction content, interaction theme, and some dynamic characteristics. We then analyze and classify situation aware user behaviors in social networks using our proposed semi–supervised clustering detection method. This allows us to promptly detect malicious social bots using only a small number of tagged users.

SOFTWAREDESIGN

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities –design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the laststage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

TESTING STRATEGY

A strategy for system testing integrates system test cases and design techniques into a well–planned series of steps that results in the successful construction of software. The testing strategy must co–operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing mustaccommodate low–level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptancetesting.

White Box Testing:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language

of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the softwareworks.

Fig.1: HOME SCREEN OF OSN

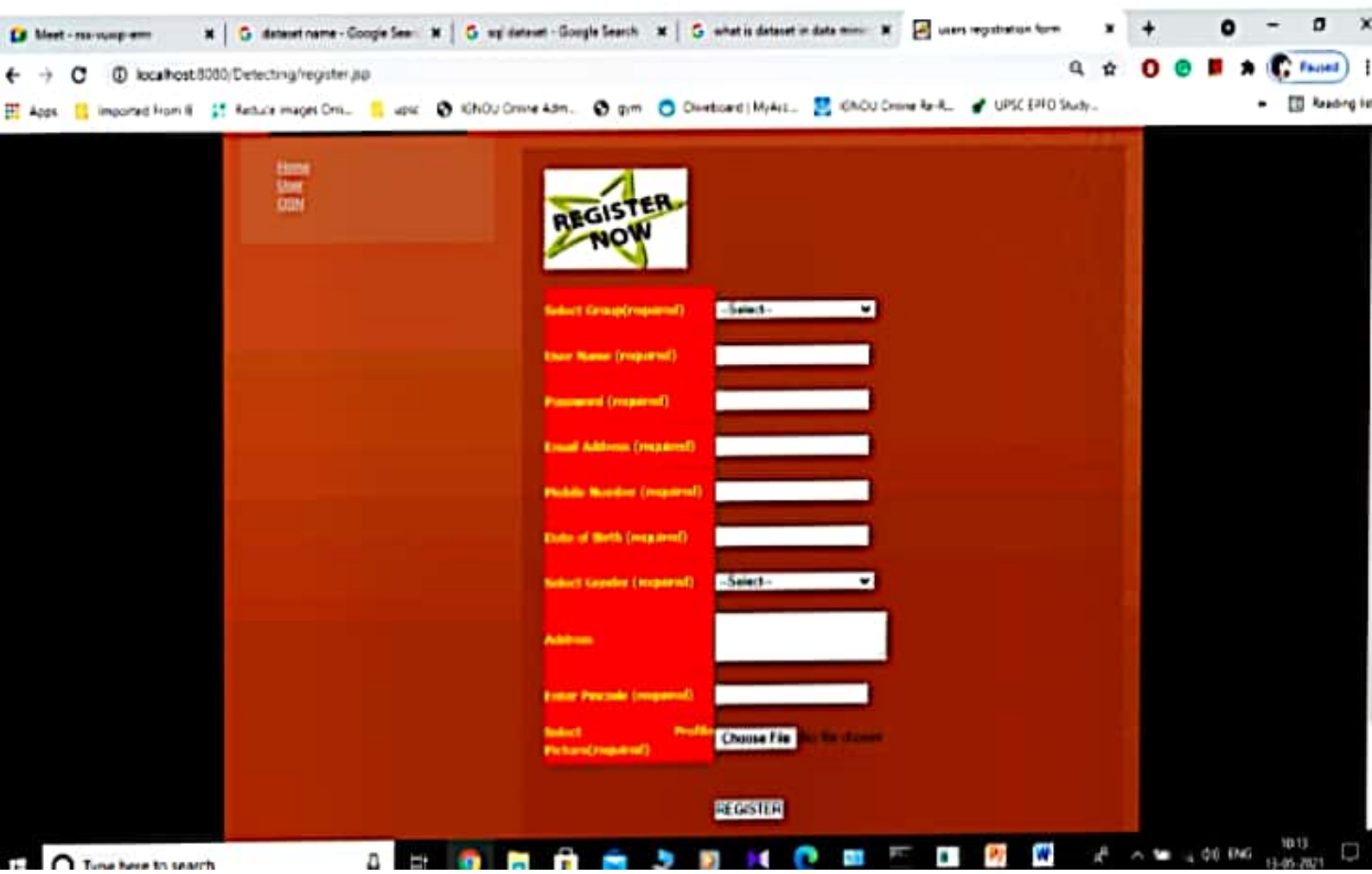


Fig.3: ALL USERS APGE

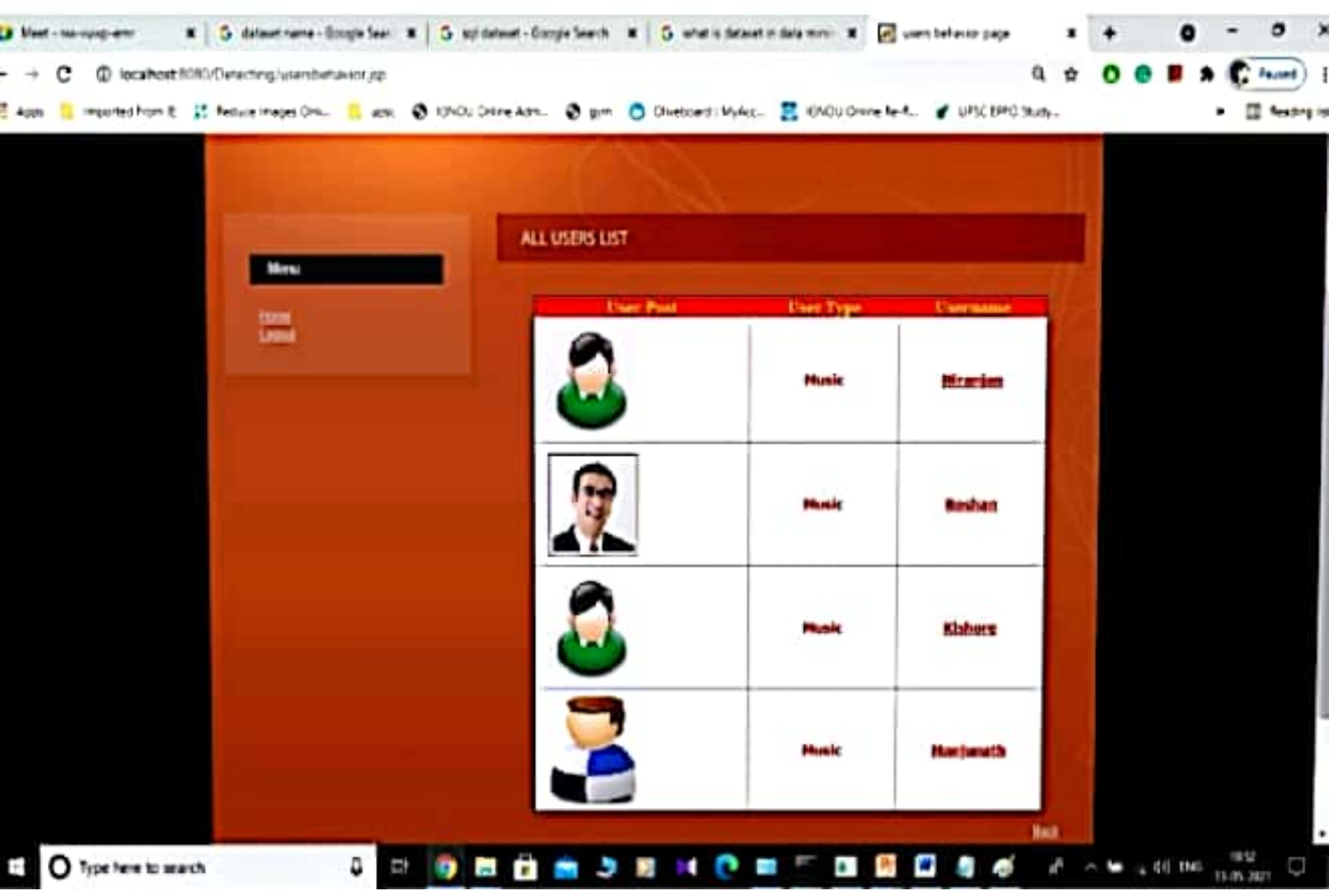


Fig.5: ADD NEGATIVE WORDS

CONCLUSION

We proposed a novel method to accurately detect malicious social bots in online social networks. Experiments showed that transition probability between user click streams based on the social situation analytics can be used to detect malicious social bots in online social platforms accurately. In future research, additional behaviors of malicious social bots will be further considered and the proposed detection approach will be extended and optimized to identify specific intentions and purposes of a broader range of malicious socialbots.

REFERENCES

F. Morstatter, L. Wu, T. H. Nazer, K. M. Carley, and H. Liu, `A new approach to bot detection: Striking the balance between precision and recall," in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining, San Francisco, CA, USA, Aug. 2016, pp.533_540.

C. A. De Lima Salge and N. Berente, ``Is that social bot behavingunethically?"Commun. ACM, vol. 60, no. 9, pp. 29_31, Sep. 2017.

M. Sahlabadi, R. C. Muniyandi, and Z. Shukur, ``DetectingabnormalBehaviorin social network Websites by using a process mining technique,"J. Comput. Sci., vol. 10, no. 3, pp. 393_402,2014.

F. Brito, I. Petiz, P. Salvador, A. Nogueira, and E. Rocha, ``Detectingsocial-network bots based on multiscale behavioral analysis," in Proc.7th Int. Conf. Emerg. Secur. Inf., Syst. Technol. (SECURWARE), Barcelona,Spain, 2013, pp.81_85.

T.–K. Huang, M. S. Rahman, H. V. Madhyastha, M. Faloutsos, andB. Ribeiro, ``An analysis of socware cascades in online social networks,"in Proc. 22nd Int. Conf. World Wide Web, Rio de Janeiro, Brazil, 2013,pp.619_630.

H. Gaoet al., ``Spam ain't as diverse as it seems: Throttling OSN spam withtemplates underneath," in Proc. 30th ACSAC, New Orleans, LA, USA,2014, pp.76_85.

E.Ferrara,O.Varol,C.Davis,F.Menczer,and A.Flammini,``Theriseofsocialbots,"Com mun. ACM, vol. 59, no. 7, pp. 96_104, Jul. 2016.

T. Hwang, I. Pearce, and M. Nanis, ``Socialbots: Voices from the fronts,"Interactions, vol. 19, no. 2, pp. 38_45, Mar.2012.

Y. Zhou et al., ``ProGuard: Detecting malicious accounts in socialnetwork–based online promotions," IEEE Access, vol. 5, pp.1990_1999,2017.

Z. Zhang, C. Li, B. B. Gupta, and D. Niu, ``Ef_cientcompressedciphertext length scheme using multi–authority CP–ABE for hierarchicalattributes," IEEE Access, vol. 6, pp. 38273_38284, 2018.doi:10.1109/ACCESS.2018.2854600.

editorstdjournal@gmail.com

Science, Technology And Development Journal

An UGC CARE APPROVED GROUP - II MULTIDISCIPLINARY JOURNAL

SCOPUS INDEXED TILL - 1998

ISSN NO: 0950-0707

Impact Factor : 6.1

This is to certify that the paper entitled

DETECTING AND REMOVING MALICIOUS SOCIAL BOTS

Authored By

Dr.M.Varaprasad Rao(Professor)

From

CMR TECHNICAL CAMPUS HYDERABAD

Has Been Published in

STD JOURNAL, VOLUME X, ISSUE VI, JUNE 2021

Francis J. Keefe, USA

Editorstdjournal@Gmail.Com

DOI: 10.15433/JES

editorstdjournal@gmail.com

Science, Technology And Development Journal

An UGC CARE APPROVED GROUP - II MULTIDISCIPLINARY JOURNAL

SCOPUS INDEXED TILL - 1998

ISSN NO: 0950-0707

Impact Factor : 6.1

This is to certify that the paper entitled

DETECTING AND REMOVING MALICIOUS SOCIAL BOTS

Authored By

T.VIJAYALAXMI

From

CMR TECHNICAL CAMPUS HYDERABAD

Has Been Published in

STD JOURNAL, VOLUME X, ISSUE VI, JUNE 2021

Francis J. Keefe, USA

Editorstdjournal@Gmail.Com

DOI: 10.15433/JES

editorstdjournal@gmail.com

Science, Technology And Development Journal

An UGC CARE APPROVED GROUP - II MULTIDISCIPLINARY JOURNAL

SCOPUS INDEXED TILL - 1998

ISSN NO: 0950-0707

Impact Factor : 6.1

This is to certify that the paper entitled

DETECTING AND REMOVING MALICIOUS SOCIAL BOTS

Authored By

B.DEVIKA

From

CMR TECHNICAL CAMPUS HYDERABAD

Has Been Published in

STD JOURNAL, VOLUME X, ISSUE VI, JUNE 2021

Francis J. Keefe, USA

Editorstdjournal@Gmail.Com

DOI: 10.15433/JES

editorstdjournal@gmail.com

Science, Technology And Development Journal

An UGC CARE APPROVED GROUP - II MULTIDISCIPLINARY JOURNAL

SCOPUS INDEXED TILL - 1998

ISSN NO: 0950-0707

Impact Factor : 6.1

This is to certify that the paper entitled

DETECTING AND REMOVING MALICIOUS SOCIAL BOTS

Authored By

A.SAHANA REDDY

From

CMR TECHNICAL CAMPUS HYDERABAD

Has Been Published in

STD JOURNAL, VOLUME X, ISSUE VI, JUNE 2021

Francis J. Keefe, USA

Editorstdjournal@Gmail.Com

DOI: 10.15433/JES