Suspicious E-Mail Detection using Keywords

Project report submitted in partial fulfillment of the requirement
for the degree of Bachelor of Technology in

Computer Science and Engineering/Information Technology

By

Priyank Agarwal (151349)

Under the supervision of

Mr. Amol Vasudeva

to

Department of Computer Science & Engineering and Information
Technology
**Jaypee University of Information Technology Waknaghat, Solan-
173234, Himachal Pradesh**

CERTIFICATE

I hereby declare that the work presented in this report entitled **"Suspicious E-Mail Detection using Keywords"** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2018 to May 2019 under the supervision of **Mr. Amol Vasudeva** (**Assistant Professor [Grade-2]**).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Priyank Agarwal (151349)

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Mr. Amol Vasudeva

(Assistant Professor Grade-2)

 Dept. of CSE

ACKNOWLEDGEMENT

It's our privilege to express our sincerest regards to our project supervisor **Mr. Amol Vasudeva (Assistant Professor Grade-2)**, for their valuable inputs, able guidance, encouragement, whole-hearted cooperation and direction throughout the duration of our project.

We deeply express our sincere thanks to our Head of Department **Prof. Dr. Satya Prakash Ghrera** for encouraging and allowing us to present the project on the topic **"SUSPICIOUS E-MAIL DETECTION USING KEYWORDS"** at our department premises for the partial fulfillment of the requirements leading to the award of B-Tech degree.

At the end we would like to express our sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

Priyank Agarwal(151349)

TABLE OF CONTENT

LIST OF FIGURES

ABSTRACT

The practice of looking at substantial previous databases so as to produce new data and relationship among them. In this paper a Decision tree in categorizing the suspected mail recognition (emails about crimes). In view of the hypothesis of fraud a suspicious email will have suspicious words and action words. The words such as attack, hijack, RDX, bomb, etc represent the suspicious and action words.

We connected this hypothesis to the mail training dataset and then connected ID3 algorithm to create the decision tree. The decision tree then is utilized to categorize the mail as suspicious or not. Specifically, we are keen on recognizing crimes from such information.

E-mail communication has become a part of everyday life for millions of people and has changed the way we work. So, it is very important to develop such system to prevent and suspect the criminal activities over the internet.

CHAPTER 1

INTRODUCTION

1.1   Introduction

E-mail (Electronic Mail) is one of the most popular, fastest and cheapest means of communication. It has become a part of everyday life for millions of people and has changed the way we work. Email nowadays can be sent/gotten to a single client or gatherings. A solitary E-Mail can spread among a great many individuals inside couple of minutes. These days, most people even can't envision the existence without email. For every one of these reasons, E-mail has turned into a generally utilized vehicle for correspondence of terrorists too. Many researchers have occurred and they all have centred in the region of counter-terrorism after the grievous occasions of 9/11 attempting to foresee terrorist intentions from deceptive communication. This additionally propelled us to contribute here.

It consists of 5 modules:

a) <u>Login</u>: This module is utilized by admin and clients to login into the email server. The right login credentials will help the client with entering into the email system.

b) <u>Registration:</u> The given page is used by the new clients. The clients must register to get the access of email system.

c) <u>Admin:</u> Admin can make use of this page according to his or her need to use all the features of email system,

d) <u>User Module:</u> With the help of this page users are allowed to send their mails, and further can check it.

e) <u>Mailing Module:</u> This module is used for performing mailing operations. It constitutes of checking the mails, composing the mails, and finally the sending the emails.

1.2 Problem Statement

1.2.1 Definition of the problem

To develop a new system for the suspicious email detection first we have to analyze the previous system and perform the analytics on difficult problems faced by the user. By Analysis of the system we can easily able to understand these types of problem and we will propose a new system through this we get the solution of these types of problems and will be able to remove them.

1.2.2 Existing System

The E-Mails are sent to authenticated users which are intended to receive. Some of the problems faced in this Existing System were:

a) Offensive and Criminal Users are unidentified.

b) Suspicious Mails weren't able to identify.

1.2.3 Proposed System

The more refined version in which suspicious users got identified and offensive mails got blocked.

a) Helps in finding out anti-social elements.

b) Provides more security to system which adapts it.

c) Also helps the intelligence bureau, crime branch etc.

1.3   Objective

"Suspicious or Malicious E-mail identification" is a sort of system where suspicious customers are recognized and hindered by deciding the words utilized in email by customers. These keywords, for example, threaten, attack, blast, and so on., are found in the e-mails which are sent by the registered client. Every one of these e-mails got blocked and further are checked by the authority and recognize the clients who sent suspicious emails.

Malicious E-mail is the term used to portray any code in any piece of a product framework or content that is expected to cause undesired impacts, security ruptures or harm to a framework. So, a new system will be developed which will identify the suspicious keywords and then block them and sent warnings to the user.

1.4   Methodology:

In this proposed system we are used association rule mining algorithm to detect the suspicious mail and block algorithm to block the E-Mails.

First, the keywords are detected from the e-mail to be classified as suspicious or not suspicious. In view of the hypothesis of fraud a suspicious email will have suspicious words and action words. The keywords are then stored in the database. The words such as attack, hijack, RDX, bomb, etc represent the suspicious and action words. The future tense signifying keywords, for example, will, should, shall, are utilized to demonstrate that the email is of the sort suspicious or not. The past tense signifying keywords, for example, were, was, had are utilized to demonstrate that the mail is of informative or normal type.

We will recognize the suspicious e-mails sent from the clients who are as of now registered on this site. Firstly, new clients join themselves on the site to send the e-mails to those clients who are already registered and after that see the messages from the registered clients.

UNICODE Algorithm can be used by admin to encrypt the messages sent to the users or sent some warnings about the other user's suspicious activity. In the following, suspicious words dictionary is utilized to detect the suspicious words which are not really utilized in the typical normal e-mails or information exchange.

1.5   Organizations:

Chapter 1: Includes the brief introduction of the project to give the basic idea what has to be implemented.

Chapter 2: Includes the literature review from numerous journals, papers from various websites and conferences.

Chapter 3: Includes the system design, tools and techniques used for the implementation of the project.

Chapter 4: Includes the performance analysis and results.

Chapter 5: Includes the conclusion, limitations and future enhancements of the project.

CHAPTER 2

LITERATURE SURVEY

The exploration in the region of email analysis and suspicious messages typically centers around two regions to be specific: email traffic analysis and email content analysis.

2.1-   [1]Modeling Suspicious Email Detection using Enhanced Feature Selection by (Sarwat Nizamani, Nasrullah Memon, Uffe Kock Wiil, Panagiotis Karampelas)

In this, they introduced a suspicious email location display which consolidates upgraded highlight choice. They proposed the application of highlight choice methodologies alongside characterization strategy for psychological militant's email discovery. The shown model spotlights on the assessment of ML calculations, for example, decision tree (ID3), strategic relapse, Support Vector Machine (SVM) and Naïve Bayes (NB) for recognizing messages containing suspicious keywords. In the accompanying writing, different calculations accomplished great exactness for the coveted assignment. Be that as it may, the outcomes accomplished by those calculations can be additionally enhanced by utilizing fitting component determination systems.

They utilized ML strategies to recognize the suspicious messages. It assesses the execution of four classifiers with the component determination techniques. As the essentialness of FSS include determination procedure in the errand of email arrangement has been recognized, the following subsection examines it diagnostically.

Featuring Selection (FS) is a means to deal with pick a fragment of the principal component space. The measure of highlights in the space impacts the calculation time and in addition the precision of the classifier. The key thought behind component assurance is to glance through a possible subset of highlights by reviewing them, through a few evaluators. In this paper they concentrated on real component assurance by which

they could accomplish unassumingly better execution of the required errand even with the present figurings.

2.2-   [2]The Detection of Suspicious Email Based on Decision Tree
(Samruddhi Rane, Gargi Moholkar , Kajal Yerunkar , Prof. Nilima Nikam)

The following study defines that the suspicious or offensive email words can be detected by Decision Tree and the whole mechanism is developed in Microsoft Visual Studio 2010 using ID3 Classification Algorithm. MySQL Server has been utilized as a backend and a data record has been made for the planning test set of messages. With some rules, a new decision tree is constructed which help in deciding the mail is suspicious or not. This application is stage autonomous and it is a single tick application which once realized through Visual Studio does not require any kind of programming. It's basic and easy to understand.

The proposed structure by the analysts at first thinks some features, for instance, "suspicious watchwords" and "non-suspicious catchphrases" from the email. Then, blend of "suspicious watchwords" and "non-suspicious catchphrases" is broke down. In case suspicious watchwords are accessible in an email with no non-suspicious catchphrases, the email will be distinguished as suspicious and the danger of a potential future fear based oppressor events will be reflected. On the off chance that some suspicious watchwords are contained alongside non-suspicious catchphrases, the email will be in addition named "might be-suspicious" in light of the way that it may be the situation of an email in which individuals discusses the past occasions, perhaps for empathy, sympathy, etc. In this instrument, the highlights will be confined near to the unique condition.

ID3 Decision tree calculation has been used to arrange the records. Estimation starts with a testset T = {email1, email2, email3… emailn} and classified as Suspicious = {Yes, No, Maybe}. Each email is given a name. The reason behind existing is to build up a mechanical assembly that watches the models from the course of action test set and can bundle another test email as suspicious, non-suspicious or might be-suspicious.

Calculation evacuates the Suspicious Keywords and the Non-Suspicious Indicators which were accesible in the arranging test set of messages.

The Proposed work (utilizing keyword extraction and catchphrase trait called tense) will be useful for recognizing the misleading email and to get data so as to take powerful activities to lessen criminal exercises.

2.3- [3]The Suspicious E-Mail Detection via Triple DES Algorithm: Cryptography Approach (Nisha Rani, Mrs. Neetu Sharma)

In this paper, they have associated Cryptography frameworks to recognize suspicious messages, that is, a mail that alerts of anticipated mental oppressor events. They have associated Triple DES (Data Encryption Standard) estimations, underscoring at first on In an email, the fundamental key is utilized to DES-encode the message. The second key further is connected to DES-decipher the mixed email. The twice-mixed email is then encoded again with the central key to get the last figure content. So, three-advance technique is known as triple-Triple-DES is just DES finished on different occasions with 2 keys utilized in a particular solicitation. (Triple-DES should moreover be conceivable with 3 separate keys instead of only 2.)

In this material, we will distinguish the suspicious sends sent from the clients who are as of now enrolled on this System. Initially new clients join themselves on the site to send the sends to those clients who effectively enlisted and after that see the messages from the enrolled clients. Triple DES Algorithm utilized by administrator to scramble the messages sent to the clients or sent a few alerts about the other clients' suspicious movement.

Administrator is made the information lexicon of suspicious words and this information word reference makes help to recognize the suspicious action of the clients. Administrator further will be included the suspicious words into the current Suspicious Words information lexicon.

Triple DES Algorithm utilized by administrator to scramble the messages sent to the clients or sent a few admonitions about the other client's suspicious activity. In this proposed work, suspicious words lexicon is utilized to identify the suspicious words which are not really utilized in the ordinary informing or correspondence.

2.4- **[4]The Suspicious E-Mail Detection via Decision Tree: A data mining approach (S. Appavu alias Balamurugan and Ramasamy Rajaram)**

In this, Decision tree is applied in speculated email revelation (messages about criminal activities). Precariousness speculation recommends that deceptive creation is portrayed by diminished repeat of first individual pronouns and prohibitive words and raised repeat of negative inclination words and action activity words. We related this model of perplexity to the approach of email dataset, by then related ID3 figuring to settle on the choice tree. The choice tree that is made is utilized to test the email as suspicious or not. Specifically, we are captivated in recognizing flabbergasting and perhaps crimes from such information.

A data mining-based decision tree computation to recognize email regarding criminal activities. It is made expressly to recognize deluding correspondence in email.

Association chief is utilized to provide the relationship between the email sender and the Processing center. The Content channel expect the basic occupation i.e., it uses the pre-preparing and organizing computation, for example, Decision tree, etc to disconnect the suspicious messages. This yield is passed on to the specialist with the help of Delivery overseer.

Email pre-handling incorporates the method of changing the arrangement dataset into a depiction suitable for the choice tree – ID3 computation. This stage removes the suspicious words from the data set. It includes the going with two phases:

    a) Evacuation of non-discriminative words

b) Suffix Striping

Then the appropriate association rules are made in accordance to the training dataset by extracting features from the input text which can classify the e-mail into suspicious or informative or normal.

The explained work will be valuable for perceiving the deceptive email and will in like manner help the agents to get the information in order to take incredible exercises to decrease wrongdoings.

CHAPTER 3

SYSTEM DEVELOPMENT

3.1   System Architecture:

3.1.1   Two-Tier and Three-Tier architecture:

- Two-Tier Architecture: The two-level depends on Client Server design. The two-tier architecture resembles customer server application. The immediate correspondence happens among customer and server. There is no moderate among customer and server. As a result of tight coupling a 2 tiered application will run quicker.

Fig. 3.1(2-Tier Architecture)

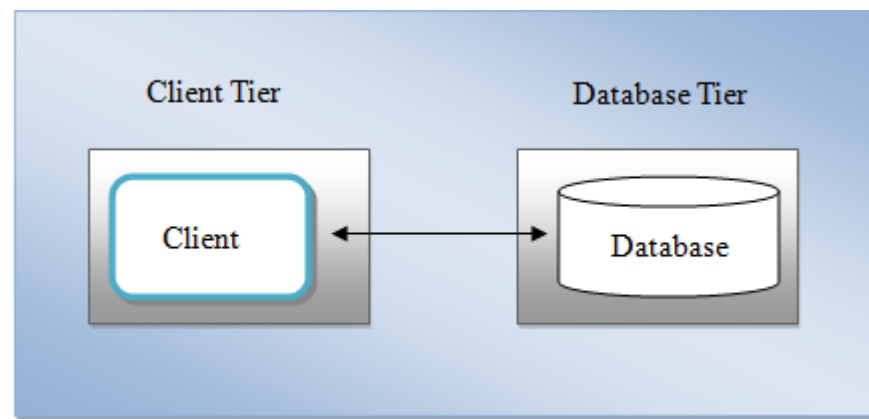- Three-Tier Architecture: A three-tier engineering is a customer server design in which the useful procedure rationale, information get to, PC information stockpiling and UI are created and kept up as free modules on discrete stages. Three-level engineering is a product configuration design and an entrenched programming engineering.
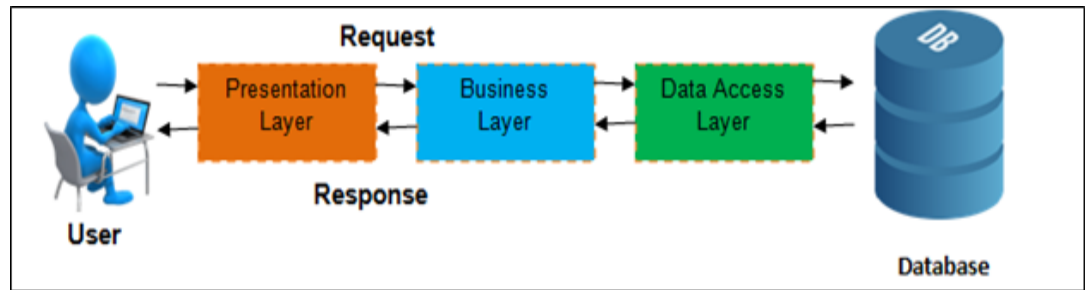
Fig. 3.2(3-Tier Architecture)
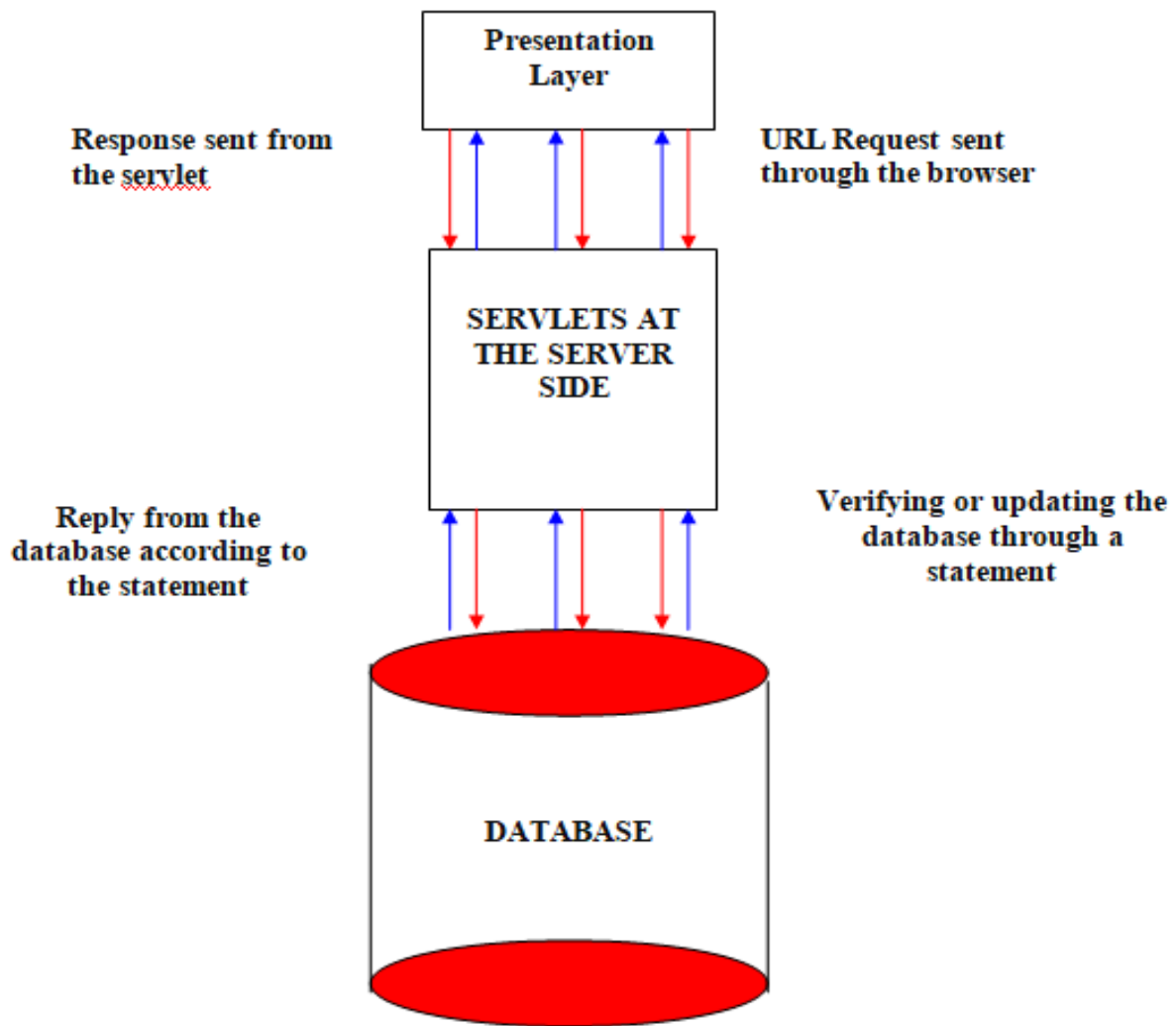
Fig. 3.3(URL Pattern)

3.1.1.1   **Advantages of Tiers:**

- Layering makes one enhance practicality of the code, advance the manner in which that the system works when passed on in distinctive ways, and give a sensible framework between zones where some development or structure choices should be taken.

- Setting layers on discrete physical levels could help execution by dispersing the stack over various servers. It could in like manner help with security by isolating progressively delicate portions and layers onto particular frameworks or on the Intranet versus an internet.

3.1.1   **MVC Architecture:**

In Model 1, an intrigue is made to a JSP or servlet and after that JSP or servlet handles all requests concerning the intrigue, including setting up the demand, favoring data, dealing with the business justification, and delivering a response. The Model 1 design is usually utilized in littler, straightforward undertaking applications because of its simplicity of advancement.

Likewise, the Model 1 engineering superfluously integrates the business rationale and introduction rationale of the application.
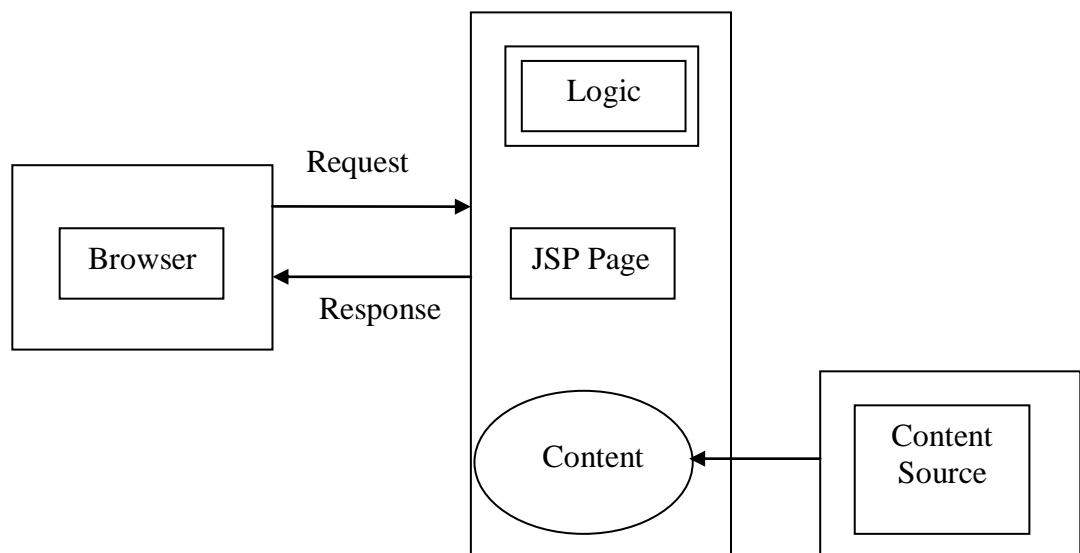
Fig. 3.4 MVC (Model-1)

Model 2 is a setup configuration utilized in the structure of Java Web applications which limits the part of substance from the premise used to get and control the substance. As Model 2 drives a section among reason and show, it is regularly associated with the model-view-controller (MVC) point of view. While the right kind of the MVC "Represent" was never showed up by the Model 2 plan, different streams endorse a formalized layer to contain MVC Model code. The Java Blue Prints, for example, at first embraced utilizing EJBs to embody the MVC Model.

In a Model 2 application, every one of the requests from the customer program are sent to the controller. The controller plays out any support essential to verify the right substance for show off. It by then puts the substance in the context (everything considered as a Java Bean or POJO) and picks which view it will pass the data to. The view by then renders the substance sent by the controller. Model 2 is held onto for medium similarly concerning liberal assessed applications.
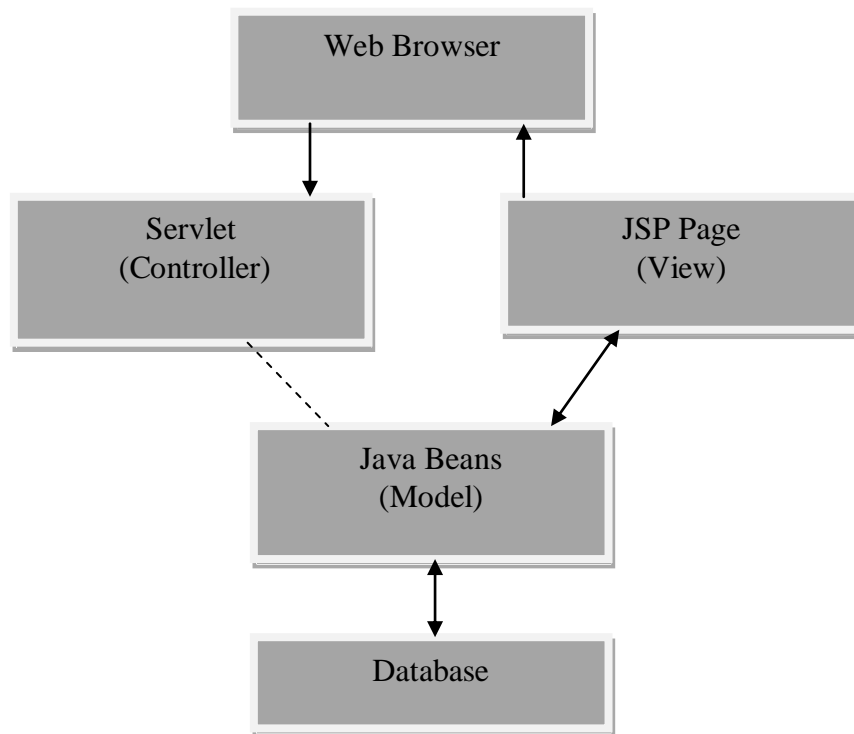
Fig. 3.5 MVC (Model 2)

3.1.2.1 Advantages of MVC:

It is an arrangement guide to make web applications in which three fragments are made known as Model, View and Controller so that one can easily develop the application. Since there are different layers addressing methods of reasoning. Distinctive methods of reasoning there will be ideal division of jobs or justifications.

- Alteration of one rationale does not influence different rationales.
- Simple to keep up and upgrade the undertaking.
- Parallel improvement is conceivable because of this efficiency is great.

3.2   Software Process Model:

3.2.1   Agile Software Development v/s Waterfall Model
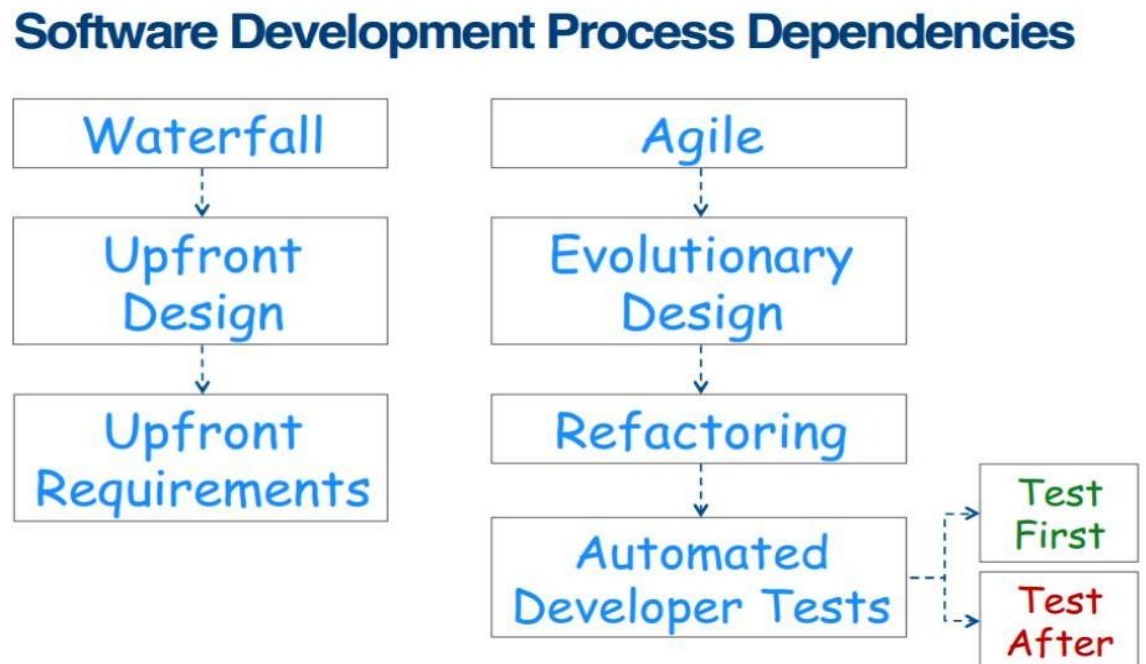
Fig 3.6 Agile Software Development v/s Waterfall Model

3.3   System Design:

3.3.1   Design Description:

Configuration is basically a blue print or it goes about as an extension between the prerequisite detail and the last answer for fulfilling the necessities.

In view of the work process portrayed above we can make the accompanying determinations for the Software System that must be produced:

• The System should be an online framework with the goal that it permits the administrator and customers to get to the protected mail over the Internet.

• Being an online framework, it empowers the clients to send messages to different clients who are as of now enrolled. An additional favorable position is since the email is conveyed immediately; there could be moment reactions from the administrator if any suspicious messages are distinguished.

• The entire process relies upon correspondences between administrator and the clients. On the off chance that every one of these interchanges are done through an online framework, the day and age for the entire procedure can be impressively cut down.

• The System needs to store the points of interest of the considerable number of clients.

• The System needs to store the subtle elements of all the data (sent sends, created sends and so forth) held by every one of the clients.

• The System needs to store the points of interest of the considerable number of necessities held by the distinctive clients.

• Since it is an electronic framework, a Login approval ought to be given so that Admin and clients will have the capacity to query and utilize alternatives that are explicit to them.

• The System ought to enable the users to input their points of interest.

• The System ought to give a choice to produce a client Report.

• The System ought to give an alternative to create square sends Report.

• The System ought to give a choice to produce chosen clients Report.

3.3.2 Top Down Approach vs Bottom Up

There are two procedures for structure up any database, the best down strategy and the base up strategy. While these systems show up radically extraordinary, they share the common target of joining a structure by depicting most of the relationship between the techniques. We should take a gander at every system.

The best down strategy begins from the general and moves to the unequivocal. In a general sense, you begin with a thought of what are required to prepare the structure and after that ask the end-clients what information they want to store. The analyst then consults with the clients to make sense of what information ought to be stored in the database. The best down strategy comparably could have insufficiencies. From time to time, top-down course of action can incite prohibited outcomes in light of the way that the master and end-clients could leave something that is essential and is critical for the structure.

The base up procedure starts with the express central indicates and moves towards the general point of view. The system analyst will research the majority of the interfaces that the framework has, screens, checking reports, and structures. The analyst work in reverse over the structure to understand what information ought to be verified in the database.

3.3.3 UML Modeling: This permits the thing fashioner to give an examination show utilizing the displaying approval that is managed by a game-plan of syntactic semantic and commonsense norms. An UML structure is tended to utilizing five unique perspectives that portray the framework from particularly interchange purpose of perspective. Each view is portrayed by a strategy of outline, which is as per the going with.

- **User Model View:**
  a) Here we can see the system as per the users.
  b) The examination depiction portrays a utilization circumstance from the end-customer's perspective.

- **Structural Model View:**
  a) Illustrate a class or the static structure of a framework at a specific point in time.

- **Implementation Model View:**
  a) In this the basic and social as parts of the framework are spoken to as they are to be manufactured.

- **Behavioral Model View:**
  a) It addresses the dynamic of social as parts of the system, portraying the coordinated efforts of gathering between various fundamental segments delineated in the customer show and essential model view.

- **Environmental Model View:**
  a) In this the basic and conduct parts of nature in which the structure is to be actualized are spoken to.

UML is explicitly created through two unique spaces:

- o UML Analysis displaying which spotlights on the user show and basic model perspectives of the structure.
- o UML design modeling, which focuses on the behavioral modeling, execution demonstrating and ecological model perspectives.

3.3.4   Class Diagrams

3.3.4.1 Class Diagram for System
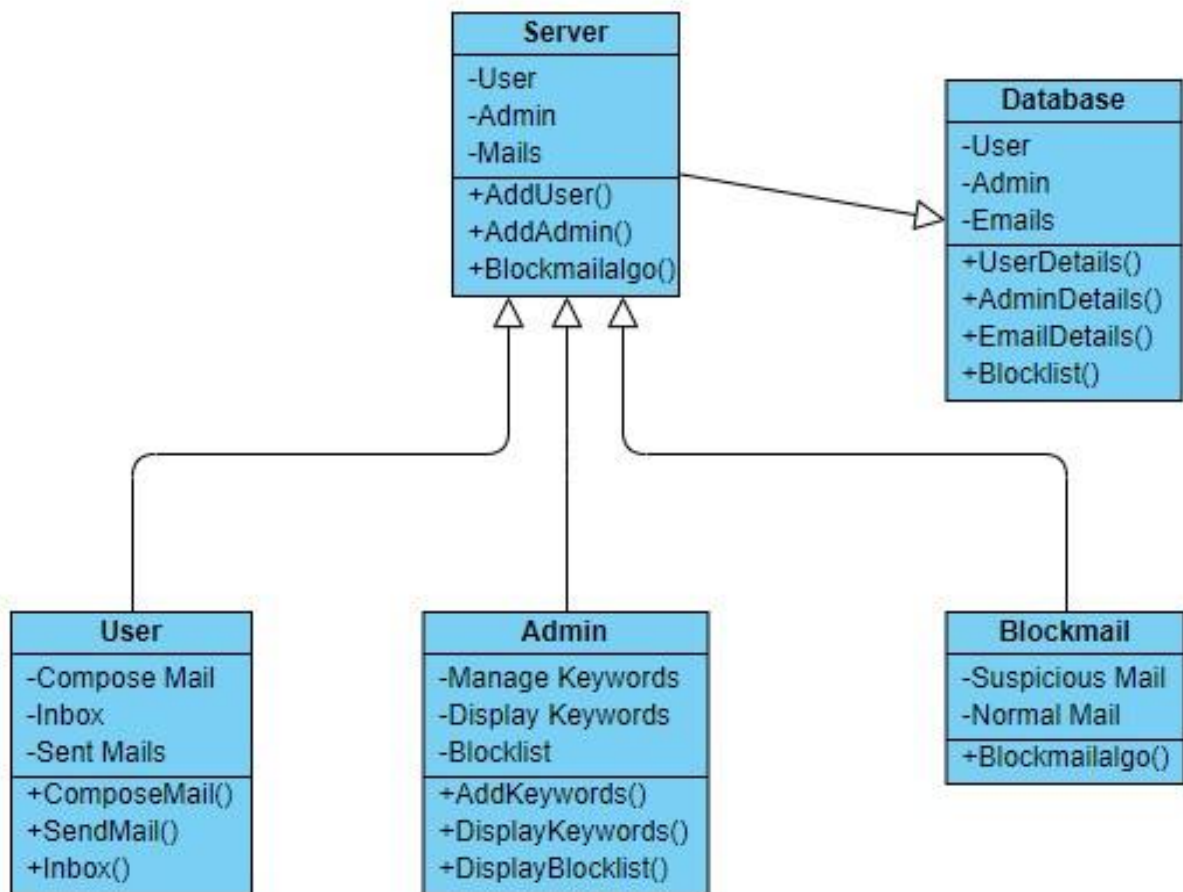
Fig. 3.7 System Class Diagram

3.3.4.2 Admin
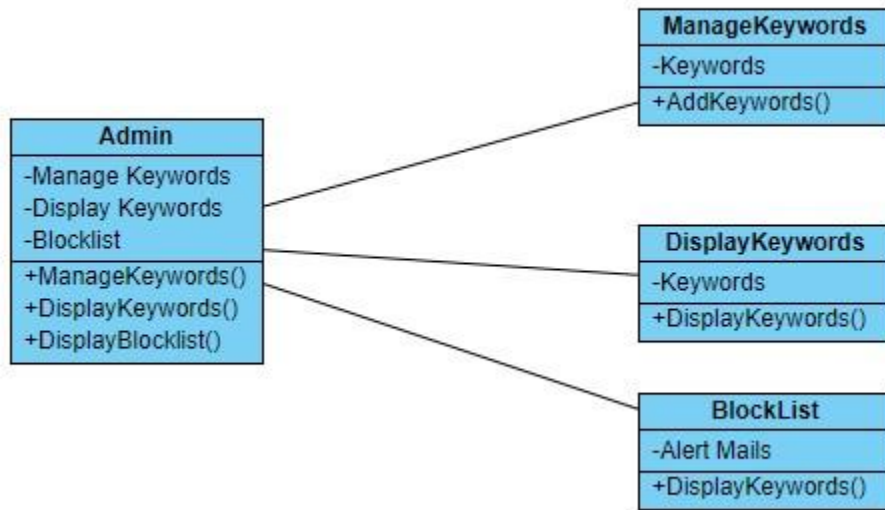
Fig. 3.8 Admin Class Diagram

3.3.4.3 User
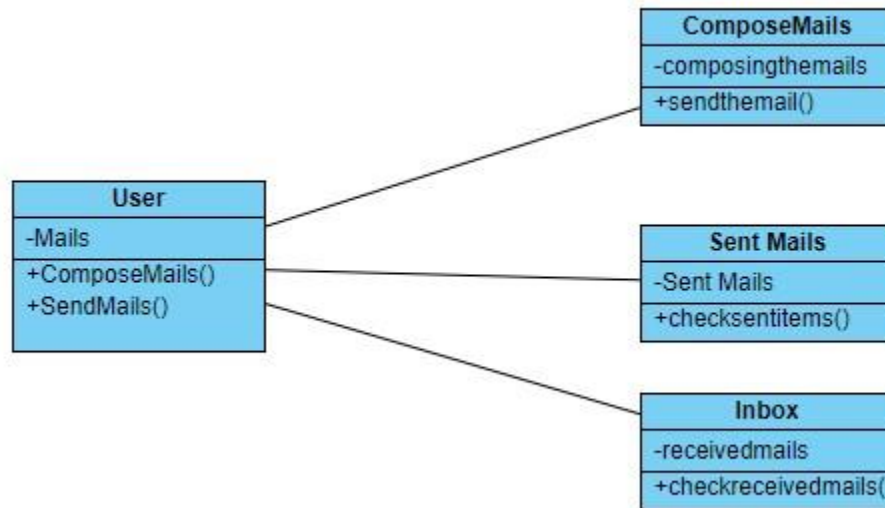
Fig. 3.9 User Class Diagram

3.3.5   Sequence Diagrams

3.3.5.1 Admin
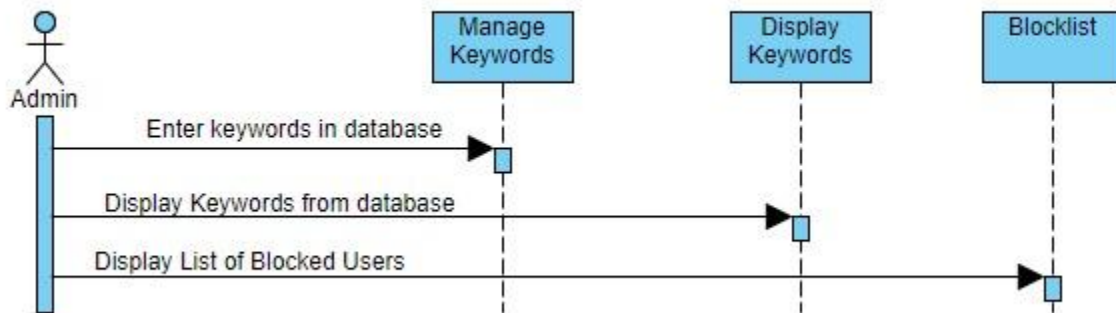
Fig. 3.10 Admin Sequence Diagram

3.3.5.2 User
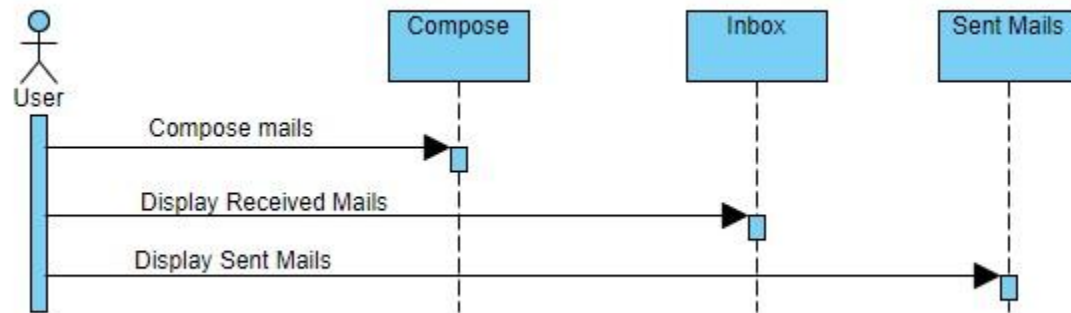
Fig. 3.11 User Sequence Diagram

3.3.6   Use Case Diagrams

3.3.6.1 Use Case for Application
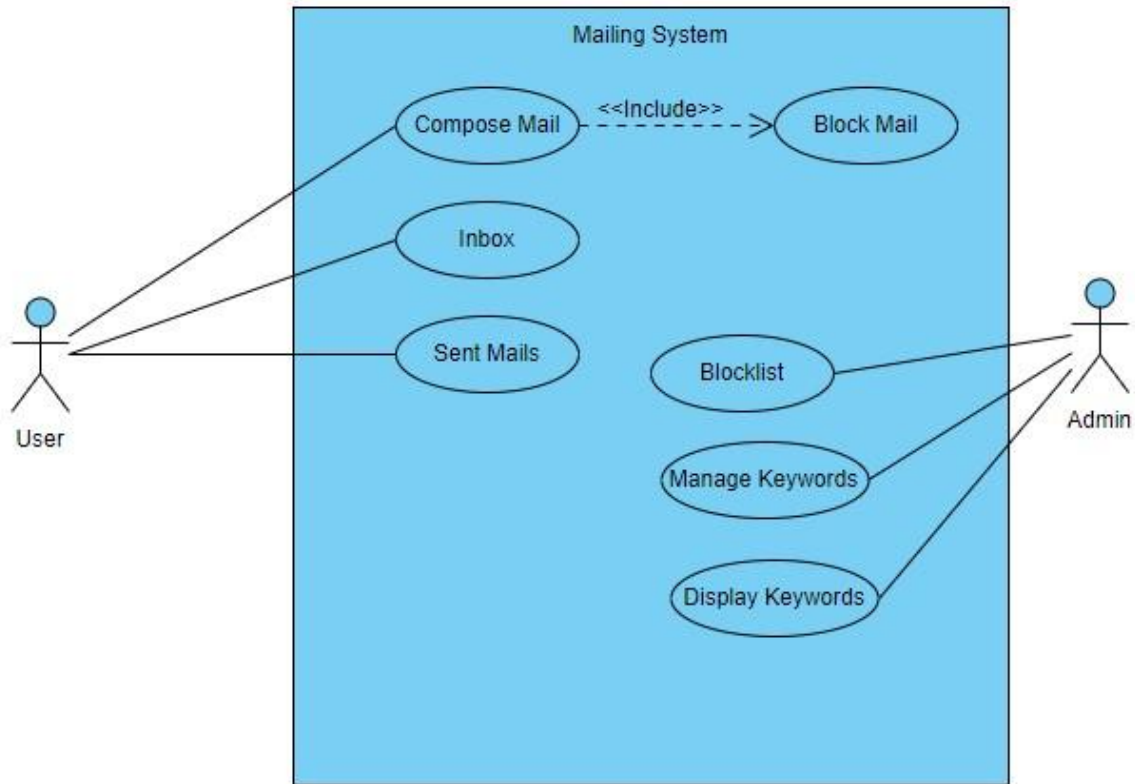
Fig. 3.12 System Use Case

3.3.6.2 Admin
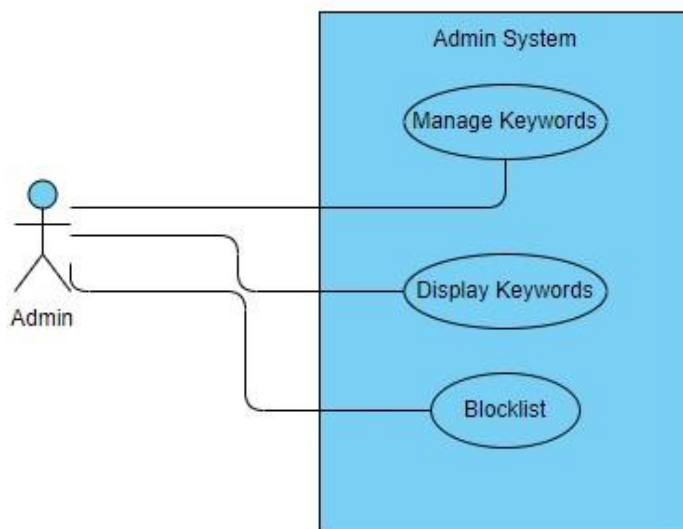
Fig. 3.13 Admin Use Case

3.3.6.3 User
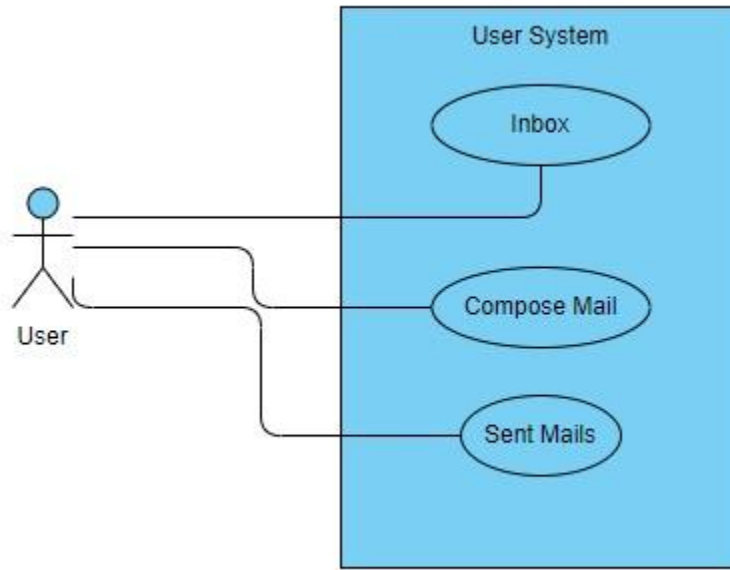
Fig. 3.14 User Use case

3.4 High Level Design:

Data Design

Table: USERINFO

Description: This table is utilized to keep up the registered client's data.

SL.NO	VARIABLE NAME	DATA TYPE	DESCRIPTION
1	EMAIL	Varchar(30)	This is the email of the user
2	PASSWORD	Varchar(20)	This the password of the user
3	FNAME	Varchar(10)	The first name of the user
4	LNAME	Varchar(10)	The last name of the user
5	DOB	Varchar(10)	The date of birth of the user
6	GENDER	Varchar(10)	The sex of the user

7	PHONE	Varchar(15)	The phone no of the user
8	CITY	Varchar(10)	The city of the user
9	STREET	Varchar(10)	The street of the user
10	COUNTRY	Varchar(10)	The country of the user
11	NATIONALITY	Varchar(10)	The nationality of the user

Table: BLOCKLIST

Description: This table is utilized for customer's mailing data.

SL.NO	VARIABLE NAME	DATA TYPE	DESCRIPTION
1	MAIL TO	Varchar(50)	To whom the user wants to send the mails
2	MAIL FROM	Varchar(50)	From whom the users got the mails.
3	SUBJECT	Varchar(100)	The subject present in the mail.
4	BODY	Varchar(500)	The message or the data present in the mail.

Table: SWORDS

Description: This table comprises of the keywords of the mailing framework.

SL.NO	VARIABLE NAME	DATA TYPE	DESCRIPTION
1	WORDS	Varchar(20)	The suspected keywords of the mails.

Table: TENSE

Description: This table comprises of the keywords that are used to extract feature from e-mail.

SL.NO	VARIABLE NAME	DATA TYPE	DESCRIPTION
1	WORDS	Varchar(20)	The suspected keywords of the mails.
2	VALUE	Varchar(20)	The tense of the text of mail.

Table: INBOX

Description: This table consists of mails that are in the inbox.

SL.NO	VARIABLE NAME	DATA TYPE	DESCRIPTION
1	MAIL TO	Varchar(50)	To whom the user wants to send the mails
2	MAIL FROM	Varchar(50)	From whom the users got the mails.
3	SUBJECT	Varchar(100)	The subject present in the mail.
4	BODY	Varchar(500)	The message or the data present in the ma

Table: MAIL

Description: This table stores the mail temporary during the e-mail detection algorithm runs.

SL.NO	VARIABLE NAME	DATA TYPE	DESCRIPTION
1	MAIL TO	Varchar(50)	To whom the user wants to send the mails
2	MAIL FROM	Varchar(50)	From whom the users got the mails.
3	SUBJECT	Varchar(100)	The subject present in the mail.
4	BODY	Varchar(500)	The message or the data present in the ma

Table: ADMIN

Description: This table is utilized to keep up the registered admin data.

SL.NO	VARIABLE NAME	DATA TYPE	DESCRIPTION
1	EMAIL	Varchar(30)	This is email of the admin.
2	PASSWORD	Varchar(20)	This is password of the admin.

3.5   Technologies used:

3.5.1   JAVA

Java is relatively founded on C++ sentence structure, and is meant to be Object-Oriented design of java is halfway between a deciphered and accumulated dialect. The programs written in java are converted into byte codes which are secure and convenient over various platforms.

JVM checks the byte codes for uprightness accessible for all OS and the byte codes into machine explicit guidelines at runtime only.

3.5.1.1 Features

- JAVA is dispersed in its methodology and utilized for web programming.
- Java is object-oriented language which supports inheritance, encapsulation, dynamic binding and polymorphism yet does not support multiple inheritance. Each - thing in java is an object except some primitive datatypes.
- Java is strong, high performing, verified and dynamic in nature.

3.5.1.2 Significance of JAVA to the web

- The significance of java on the web is on the record of that; java broadens the Universe of objects that can move about transparently in the web. In a system, two catagories of objects are exchange between the server and the PC. They are dormant dynamic and information projects during the zones of probability and security. So java help in handling these worries which opened a way to a new program known as Applet.

3.5.1.3 Architecture

- Java structure gives a portable, robust, high performing condition for development. Java is a dynamic framework, ready to load code when required from a machine in a similar room or over the planet. Java gives portability by accumulating the byte codes for the JVM, which is then translated on every stage by the run-time condition.

3.5.1.4 Compilation of code

- The java code is converted into a machine code (known as byte code) for the Java Virtual Machine (JVM). The JVM should execute the byte code which solves the issue of portability. The code can be translated on all other machines once the code is composed and compiled on the Java Virtual Machine.

3.5.1.5 Applets and Applications

An application can be characterized as a program that keeps running on our PC under the OS of that PC. It is on a very basic level like one making using C or C++. Java's ability to make Applets makes it vital. An Applet is an application, intended to be transmitted over the Web and executed by a Java-perfect program. An applet can be characterized as a little Java program, downloaded over the pc, much equivalent to a picture. Regardless, what matter is, it is a shrewd program, not just a media report. It might be react to the customer demands or inputs and react as needs be.

3.5.2 SERVLETS:

A servlet is called a Java program that keeps running on a Web server. It is like an applet, yet is handled on the server rather than a customer's machine.

A servlet takes after exclusive server augmentation; anyway it keeps running inside a JVM on the server, so it is ensured and compact servlets work inside the territory of the server.

In servlets, separate threads inside web server process handle all servlets. This implies servlets are on the whole effective and scalable. Servlets are compact; both over OS and furthermore over web servers. They offer the most ideal platform for the development of web applications.

Servlets can be used instead of CGI substance on a web server as they can expand the server, for example, a email server that grants it broaden its value by doing virus check for every single report or overseeing mail filtering tasks.

Servlets fill in as dynamically loadable, platform independent pluggable assistant byte code obejcts on the server side that can be utilized to progressively expand server-side usefulness.

For instance a HTTP servlet can be utilized to produce dynamic HTML content when one use servlets to do dynamic content one has the following advantages:

- They use the servlet API.
- They are fast and clean than CGI script.
- All the advantages of JAVA are included in it.

3.5.2.1 Attractiveness of Servlets

- Can be called progressively from inside HTML, pages utilizing server-side include tag.

- Can be easily configured using GUI-based Admin tool.

- Can be associated together, so that on servlet can call another servlet or a different servlets in course of action

- The servlet security model and servlet and box shield your framework from unpleasant conduct when they are downloaded on your system.

3.5.2.2 Servlet Advantage

One of interest of the servlets Programming interface is that they are protocol independent. It predicates nothing about the protocol being used to transmit on the net, how it is loaded, and the server environment it will keep running in. They are significant, in the way that it empowers the Servlet Programming interface to be installed in a wide scope of sorts of servers.

- It is extensible-one can inherit all your features from the base classes.

- It is simple small, and easy to use.

3.5.2.3 Features of Servlets

Some of the features of servlets are as follows:

- Servlets are fast as servlets are loaded once; they offer much better execution over their CGI counterparts.

- Servlets are persistent. Servlet are loaded just by the web server and can keep up services between requests.

- Servlet is platform independent.

- Servlet are secure

- Servlet is utilized with a variety of customer.

3.5.3 JSP

Called as Java Server Page. A fundamental Java Server Page comprises of plain content and markup and can alternatively exploit installed contents and of alternate capacities for making dynamic substance. JSP incorporates a system for characterizing dynamic qualities for custom labels. Any scripting dialect can be utilized for this reason; normally Java is actualized, however the JSP determination characterizes a custom articulation dialect planned explicitly for the assignment.

3.5.3.1 Life cycle of JSP

To understand JSP, it requires the understanding the basic life cycle that JSP pursues. The three stages of JSP lifecycle are: initialization, service, and destruction. This lifecycle ought to appear to be familiar and is very same to the one depicted for Servlets.

While a JSP follows the Servlet life cycle, the methods have different names. jspInit(), _jspService(), and jspDestroy() respectively for initialization, service and destruction. When JSP is out of servive, these phases enable a JSP to load resources, provide service to multiple customer demands, and destroy loaded resources.

JSP is created to improve the assignment of making content delivering HttpServlet objects and does as such by wiping out all the redundant parts of coding a Servlet. All JSP are intended to be utilized with HTTP and to produce dynamic content for the World Wide Web. The single JSP _jspService() method is also responsible for creating responses to every one of the seven of the HTTP methods. For most practical purposes a JSP engineer does not have to know anything about HTTP, nor anything over basic Java to code a dynamic JSP.
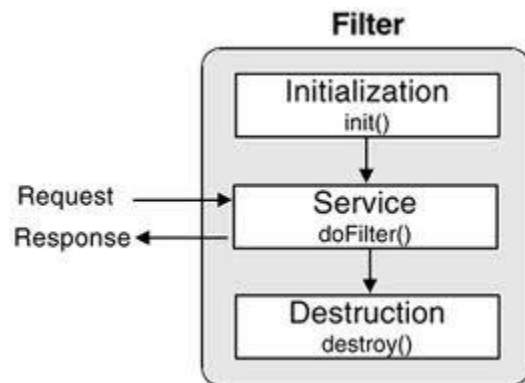
Fig. 3.15 Life Cycle of JSP

3.5.4 JDBC

The JDBC API is defined as a Java API that helps one to connect to the relational database to get any sort of tabular data. JDBC encourages one to compose Java applications that deal with following programming exercises:

- Connect to an information source, similar to a database
- Forward questions and refresh explanations to the database
- Retrieve and process the results got from the database in answer to your request.

3.5.4.1 What does JDBC do?

JDBC do the following 3 things:

- Create a connection with a database
- Forward SQL queries
- Process the outcomes

3.5.5 HTML

Hypertext Markup Language (HTML), the languages of the WorldWideWeb (WWW), enables clients to design webpages that can include text, content, illustrations, visuals and Hyperlinks.

The concept behind hypertext is one point to another point. We can explore through the data based on our interest and preference. HTML isn't a programming language yet it is

an utilization of ISO Standard 8879, SGML (Standard Generalized Markup Language), yet specific to hypertext and adjusted to the Web.

The tags provided by HTML makes the webpage looks more attractive.

HTML tags aren't case-sensitive. Utilizing graphics, illustrations, text styles, various sizes, shading, and so on can enhance the appearance of the record. Others that isn't a tag is a piece of the document itself.

3.5.5.1 Advantages of HTML

- HTML is platform independent
- A HTML document is simple and small and hence it is very easy to send it over the net. It is small as it does not include formatted information.
- HTML tags aren't case-sensitive.

3.5.5.2 Basic HTML tags

<!-- -->	Used for mentioning Comments
<HTML>…………..</HTML>	The start tag of an HTML page
<HEAD>…………..</HEAD>	Contains meta information of the document
<TITLE>…………..</TITLE>	Specifies a title for the document
<H1> to <H6>	Defines a various headings
<P>…………..</P>	Used for a paragraph
<A>…………..</A>	Creates Hypertext links.
<BODY>…………..</BODY>	Is the displayed page content
<TH>…………..</TH>	Table heading
<CENTER>…………..</CENTER>	Center the text
<TABLE>…………..</TABLE>	Creates table
<FORM>…………..</FORM>	Creates a form
<TD>…………..</TD>	Table data.
<BUTTON>…………..</BUTTON>	Creates a button
<UL>…………..</UL>	Creates unordered list
<OL>…………..</OL>	Creates ordered list
<TR>…………..</TR>	Table row
<LI>…………..</LI>	Creates list items

3.5.6 Java Script

Javascript is an object oriented PC programming language generally used to make intuitive impacts inside internet browsers. The automatic storage management in javascript is provided by the garbage collector.

For example, javascript statements written in a HTML page can perceive and react to the customer requests for instance, mouse clicks from Input, and page navigation in a user application for Navigator.

For example, you can use JavaScript function to add dynamic elements within the website, to display a popup, to get the value when user clicks a button, to validate the input entered by the user and in many more places.

With no network transmission, Java Script can interpret the input information accordingly of the customer opening or leaving a page and if the information is invalid javascript can alert the customer by displaying message box or execute an applet , play a sound or speak with a module.

3.5.6.1 Features

- JavaScript is browser dependent
- Java script is written into HTML file and is executed within them.
- JavaScript is an object-based language.

- JavaScript can be interpreted by the browser at run time.

3.5.7 PL/SQL:

PL/SQL is an augmentation to SQL. It enables us to utilize all the SQL information Manipulation articulations including embed, erase, refresh and select and additionally the exchange handling proclamations Commit, Rollback and spare point.

3.5.7.1 What can PL/SQL do?

PL/SQL can execute queries, retrieve data, insert records, update records, delete records, create new databases, create new tables, create stored procedures, create views and many more that can help user to manipulate and read data from the database.

3.5.7.2 PL/SQL commands and syntax

SQL directions are divided into a few different types, among them transaction controls and security measures, data manipulation language (DML) and data definition language (DDL) statements.

The transaction controls help manage transaction processing, guaranteeing that transactions are either finished or rolled back if mistakes or issues happen. The security statements are utilized to control database access.

The DML vocabulary is utilized to retrieve and manipulate information, while DDL statements are for defining and adjusting database structures.

SQL keywords are NOT case sensitive for example; select is the same as SELECT.

3.5.7.3 Some basic PL/SQL commands

SELECT - Extracts information from a database

UPDATE - Updates information in a database

DELETE - Deletes information from a database

INSERT INTO - Inserts new information into a database

CREATE DATABASE - Creates database

ALTER DATABASE - Modify database

CREATE TABLE - Creates table

ALTER TABLE - Modifies a table

DROP TABLE - Deletes a table

CREATE INDEX - Creates an index (search key)

DROP INDEX - Deletes an index

3.6 Algorithm Design

Numerous methods, for example, Naïve bayes, Nearest Neighbor, Support Vector Machines, Regression, Decision Trees, TFIDF Style Classifiers and Association classifiers have been created for text classification.

In this project, we use an association rule mining algorithm (Apriori calculation) to recognize suspicious email and the further order into the classification into the alert and informative or normal emails.

It is developed explicitly for identifying irregular and beguiling correspondence in email. The proposed technique is executed utilizing the Java language. In this algorithm, there are three sections: Email Preprocessing, Building the associative classifier and validation.

3.6.1   Email preprocessing

Email preprocessing is the process of transforming email messages into a portrayal suitable for the Apriori algorithm.

3.6.1.1 Email Data Extraction and presentation

This stage includes the extraction of highly emotional and actions words from the body and subject fields of the e-mail which are helpful to detect and categorize the e-mail into suspicious and informative or normal.

This includes four steps:

- Text term extraction which is the extraction of keywords from the e-mail.
- Lexical analysis which is the process of changing the input stream of characters into a stream of words or tokens.
- Stop word removal in which the words which does not represent the suspicious activities are removed from e-mail. For example, "the", "and", etc.
- Stemming which is the process of removing suffix as for example "Bomb" and "Bombing" do not make much difference while classification of e-mails into various categories.

3.6.1.2 Feature Selection

In view of the hypothesis of fraud a suspicious email will have suspicious words and action words. Thus, these words are stored as keywords and separated from the input

training dataset. The words such as attack, hijack, RDX, bomb, etc represent the suspicious and action words. The future tense signifying keywords, for example, will, should, shall, would are utilized to demonstrate that the suspicious email is of the sort alert. The past tense signifying keywords, for example, was, were, had are utilized to demonstrate that the suspicious email is of informative or normal type.

3.6.2 Building the Associative classifier

Email Classification is the way toward finding a set of models (or functions) that portrays and recognize data classes and concepts, for the purpose to utilize the model to predict class of objects whose label is unknown.

Each email message contains the suspicious words and a Tense, that are utilized to foresee an email is suspicious or not. First, we built the classification model

The training dataset contains 2 transactions of class Alert Email that have keywords Bomb/Hijack and a tense "will/may" in them, one transaction of class informative or useful email that have keywords Murdered/Condolence and a tense "was". By applying the Apriori algorithm we obtained a model that has two rules for the classification of e-mails into suspicious and informative or normal.

Affiliation Rule mining looks for association or correlation among items in a given large training data set. We create transactions where the items are the keywords from the e-mail. After e-mail preprocessing, by eliminating stop words and stemming, e-mail can be represented as di={t1…tn} as well as the category to which it belongs. The Apriori algorithm is utilized for mining frequent itemsets in transactional databases to discover frequent sets of words in the emails of the training dataset. Given the frequent itemsets and category, association rules can be prepared.

Market basket is one of the largest applications that uses association rules but we attempt to apply this algorithm to detect the suspicious and informative or normal mail and categorize accordingly. For example, itemsets for the suspected normal or informative e-mail can be {Tense=Past, Hijack=Yes, Death=True} and itemsets for the suspected alert e-mail can be {Tense=Future, Attack=Yes, Blast=Yes}. So, association rules can be constructed from the itemsets. The association rules that can be made from the training datasets used in this project are as follows:

- {Tense=Future or Present, Suspicious Keyword=Yes} → {Class=Alert Mail}
- {Tense=Past, Suspicious Keyword=Yes} → {Class=Informative or Normal}
- {Tense=Past, Present or Future, Suspicious Keyword=No} → {Class=Normal}

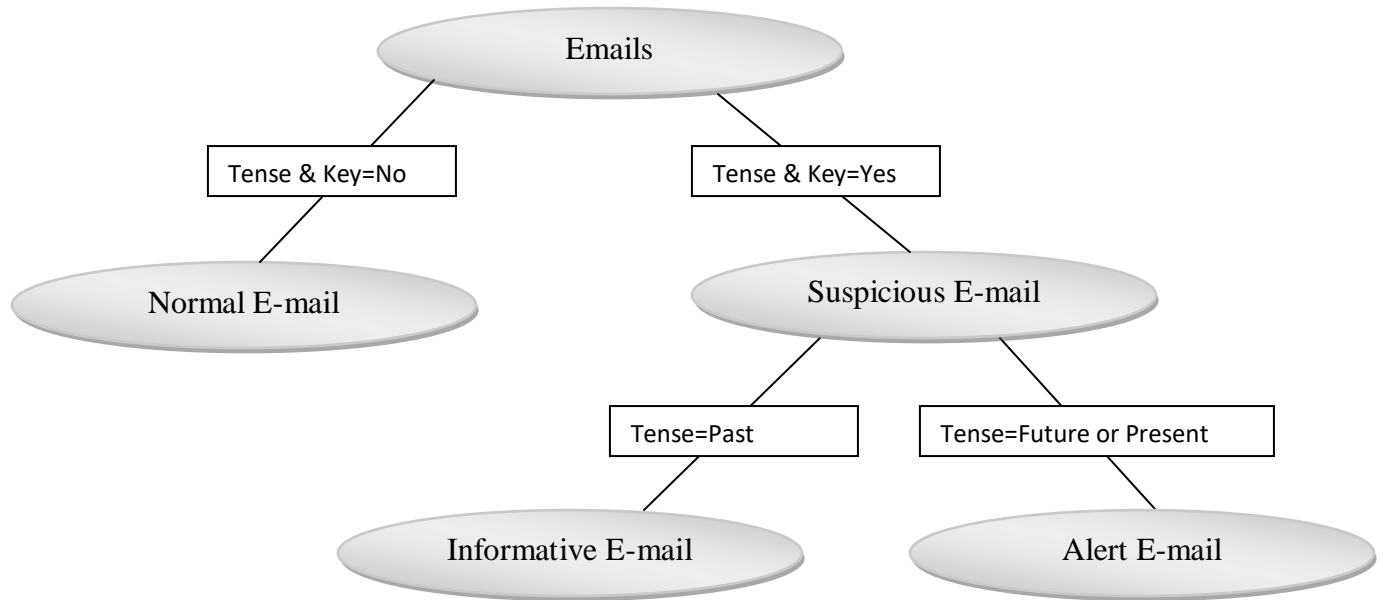The Decision tree structure to detect suspicious E-mail is shown below

Fig. 3.16 Decision tree for suspicious E-mail detection

CHAPTER 4

PERFORMANCE ANALYSIS

4.1 Earlier Developed System

The system that is known to exist to detect the suspicious mails send by the users weren't able to identify the suspicious or offensive or criminal users accurately.

4.1.1 Methodology

In the earlier developed systems they have used the keyword extraction method in which suspicious keywords are detected from the e-mail sent by the user and suspicious or offensive or criminal users can be blocked and identified and admin can take powerful actions to lessen criminal activities at appropriate time.

4.2 Proposed System

The proposed system is used to detect the suspicious mails send by the users and is able to identify the suspicious or offensive or criminal users more accurately than the earlier developed system.

4.2.1 Methodology

In this proposed system we are used association rule mining algorithm to detect the suspicious mail and block algorithm to block the E-Mails.

First, the keywords are detected from the e-mail to be classified as suspicious or not suspicious. In view of the hypothesis of fraud a suspicious email will have suspicious words and action words. Thus, these words are stored as keywords and separated from the input training dataset. The words such as attack, hijack, RDX, bomb, etc represent the suspicious and action words. The future tense signifying keywords, for example, will, should, shall, would are utilized to demonstrate that the suspicious email is of the sort

alert. The past tense signifying keywords, for example, was, were, had are utilized to demonstrate that the suspicious email is of informative or normal type.

4.3   Comparison between earlier and proposed system

The earlier system does not use the feature abstraction techniques detect the suspicious mails send by the users. So earlier developed systems cannot detect the informative mails sent by the user and cannot be blocked. For example, there are various emails in which user discusses the condolence or past events or sympathy so, these emails cannot be categorized as suspicious e-mails. But our proposed system uses the feature extraction techniques to detect such e-mails.

4.3.1   Examples from training dataset to show the comparison

4.3.3.1 "TODAY THERE WILL BE BOMB BLAST IN THE PARLIAMENT HOUSE. STOP IT IF YOU CAN."

- Earlier developed system will detect this e-mail as suspicious due to the presence of suspicious keywords bomb, blast in the e-mail.
- Proposed system will detect this e-mail as suspicious due to the presence of suspicious keywords bomb, blast in the e-mail and the future tense as it can be categorized in alert e-mail according to our association rule {Tense=Future or Present, Suspicious Keyword=Yes} → {Class=Alert Mail}

4.3.3.2 "WE ARE A GROUP OF TERRORIST WHO HAS KIDNAPPED SOME HOSTAGES. WE THREATEN TO BLOW THE BUILDING IN SESSION IF ANY IS HELD."

- Earlier developed system will detect this e-mail as suspicious due to the presence of suspicious keywords terrorist, threaten, kidnapped, blow in the e-mail.
- Proposed system will detect this e-mail as suspicious due to the presence of suspicious keywords terrorist, threaten, kidnapped, blow in the e-mail and the present tense as it can be categorized in alert e-mail according to our

association rule {Tense=Future or Present, Suspicious Keyword=Yes} →
{Class=Alert Mail}

4.3.3.3 "WE HAVE COMPLETED THE PREPARATION OF BOMB BLAST. THIS TIME
WE ARE UNSTOPPABLE AND THERE MUST BE BIG DISASTER."

- Earlier developed system will detect this e-mail as suspicious due to the
presence of suspicious keywords bomb, blast, disaster in the e-mail.

- Proposed system will detect this e-mail as suspicious due to the presence of
suspicious keywords bomb, blast, disaster in the e-mail and the present tense
as it can be categorized in alert e-mail according to our association rule
{Tense=Future or Present, Suspicious Keyword=Yes} → {Class=Alert Mail}

4.3.3.4 "THIS ATTACK IS A REACTION TO THOSE ACTIONS WHICH HINDUS HAVE
TAKEN SINCE 1947 ONWARDS. NOW, THERE SHALL BE NO ACTIONS. THERE
SHALL ONLY BE REACTIONS"

- Earlier developed system will detect this e-mail as suspicious due to the
presence of suspicious keywords attack in the e-mail.

- Proposed system will detect this e-mail as suspicious due to the presence of
suspicious keywords attack in the e-mail and the present tense as it can be
categorized in alert e-mail according to our association rule {Tense=Future
or Present, Suspicious Keyword=Yes} → {Class=Alert Mail}

4.3.3.5 "WE WERE REALLY SAD BY HEARING ABOUT THE PULWAMA ATTACK ON
14^(TH) FEBRUARY."

- Earlier developed system will detect this e-mail as suspicious due to the
presence of suspicious keywords attack in the e-mail which is not correctly
predicted

- Proposed system will detect this e-mail as informative or normal mail due to
the presence of suspicious keywords attack in the e-mail and the past tense as
it can be categorized in alert e-mail according to our association rule
{Tense=Past, Suspicious Keyword=Yes} → {Class=Informative or Normal}

4.3.3.6 "THE INCIDENCE OF 9/11 IS REALLY A BIG DISASTER. MANY INNOCENTS BECOME THE VICTIMS OF THE ATTACK. WE ALL MUSLIMS DO CONDEMN IT."

- Earlier developed system will detect this e-mail as suspicious due to the presence of suspicious keywords attack, disaster in the e-mail which is not correctly predicted.

- Proposed system will detect this e-mail as suspicious mail due to the presence of suspicious keywords attack, disaster in the e-mail and the present tense as it can be categorized in alert e-mail according to our association rule {Tense=Future or Present, Suspicious Keyword=Yes} → {Class=Alert Mail}which is not correctly predicted.

4.4   Discussion of Screens

4.4.1   This is the homepage of securemail where both admin and user can visit to login or register on the website.

Fig. 4.1 Homepage

4.4.2 This is the registration page where the users can register themselves to use the services of our website.

Fig. 4.2 Registration Page

4.4.3 This is the login module where the users can login to compose and send e-mails to other users who are already registered on our website.

Fig. 4.3 Login Page

4.4.4   This is the admin login page where admin has options to add keywords, display keywords and check the blocklist.
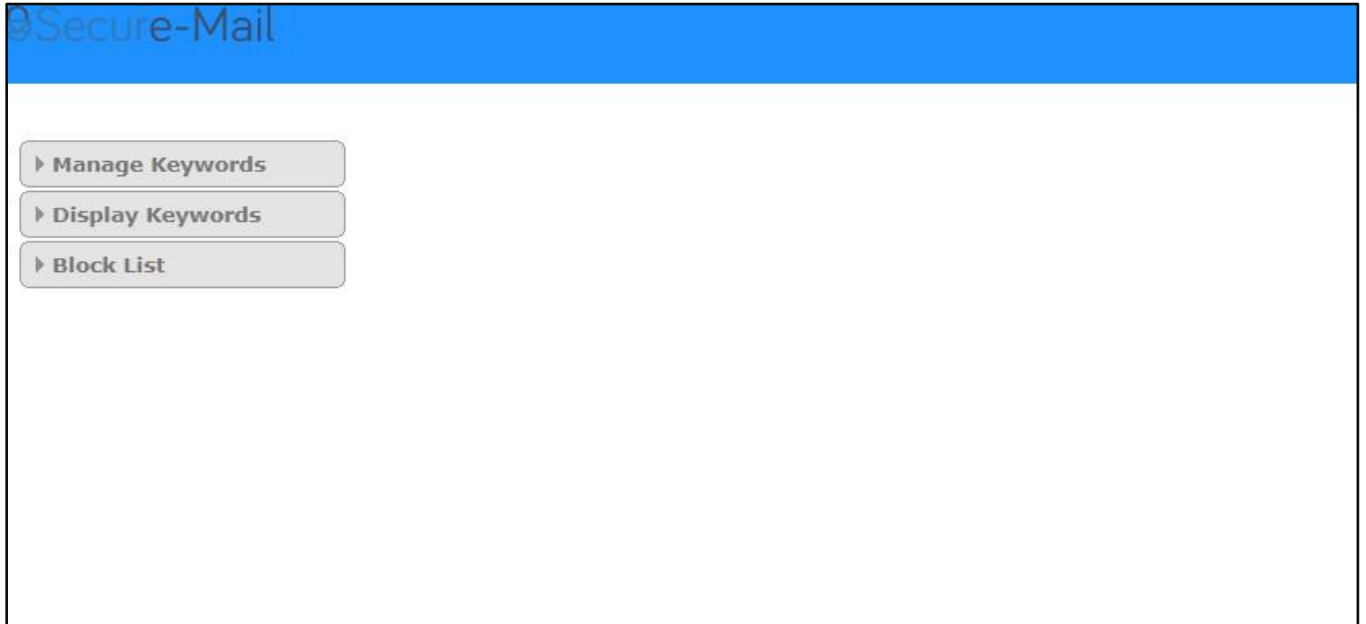

Fig. 4.4 Admin Homepage

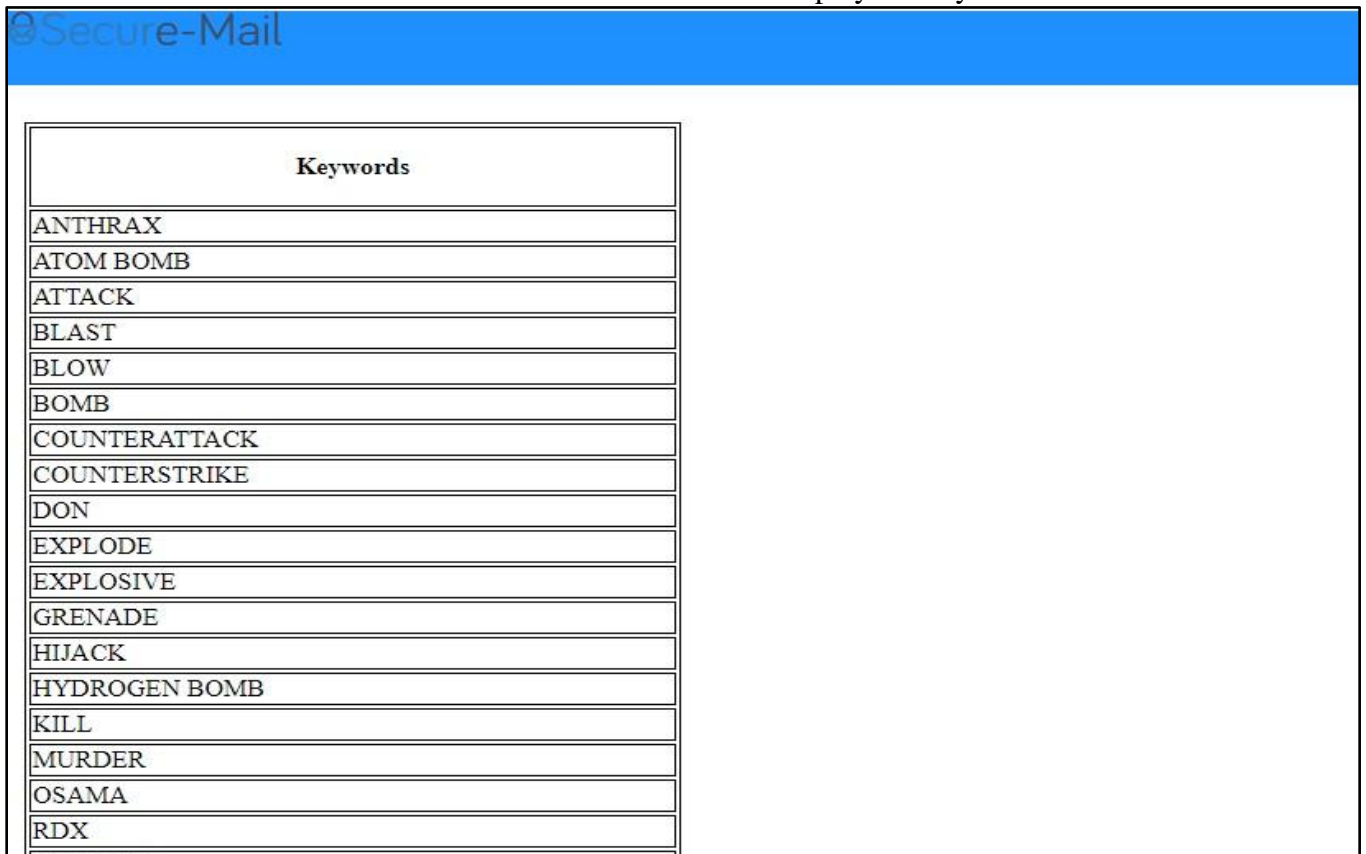**4.4.5**   This screen is shown to admin when he/she wants to display the keywords.
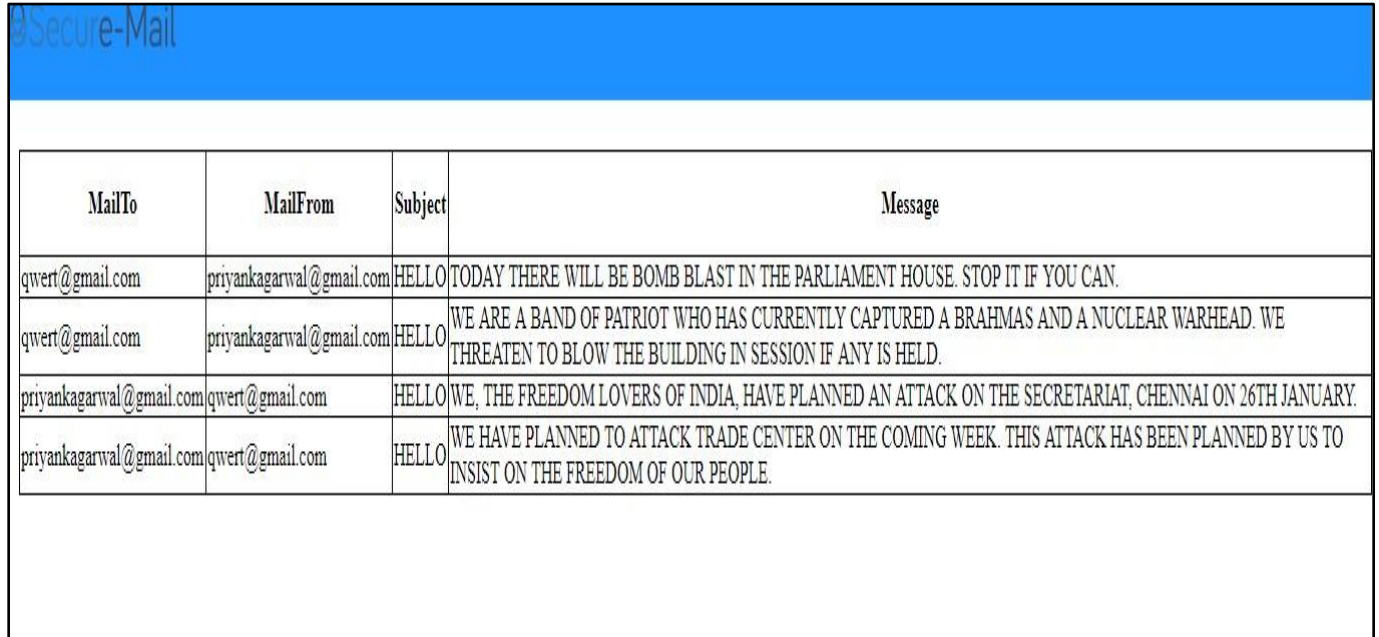

Fig. 4.5 Keywords

4.4.6   This is the list of blocked users which can be seen by the admin.

MailTo	MailFrom	Subject	Message
qwert@gmail.com	priyankagarwal@gmail.com	HELLO	TODAY THERE WILL BE BOMB BLAST IN THE PARLIAMENT HOUSE. STOP IT IF YOU CAN.
qwert@gmail.com	priyankagarwal@gmail.com	HELLO	WE ARE A BAND OF PATRIOT WHO HAS CURRENTLY CAPTURED A BRAHMAS AND A NUCLEAR WARHEAD. WE THREATEN TO BLOW THE BUILDING IN SESSION IF ANY IS HELD.
priyankagarwal@gmail.com	qwert@gmail.com	HELLO	WE, THE FREEDOM LOVERS OF INDIA, HAVE PLANNED AN ATTACK ON THE SECRETARIAT, CHENNAI ON 26TH JANUARY.
priyankagarwal@gmail.com	qwert@gmail.com	HELLO	WE HAVE PLANNED TO ATTACK TRADE CENTER ON THE COMING WEEK. THIS ATTACK HAS BEEN PLANNED BY US TO INSIST ON THE FREEDOM OF OUR PEOPLE.

Fig. 4.6 Blocklist

4.4.7   This is the login page of user where user has the options to compose, send and display the received messages.
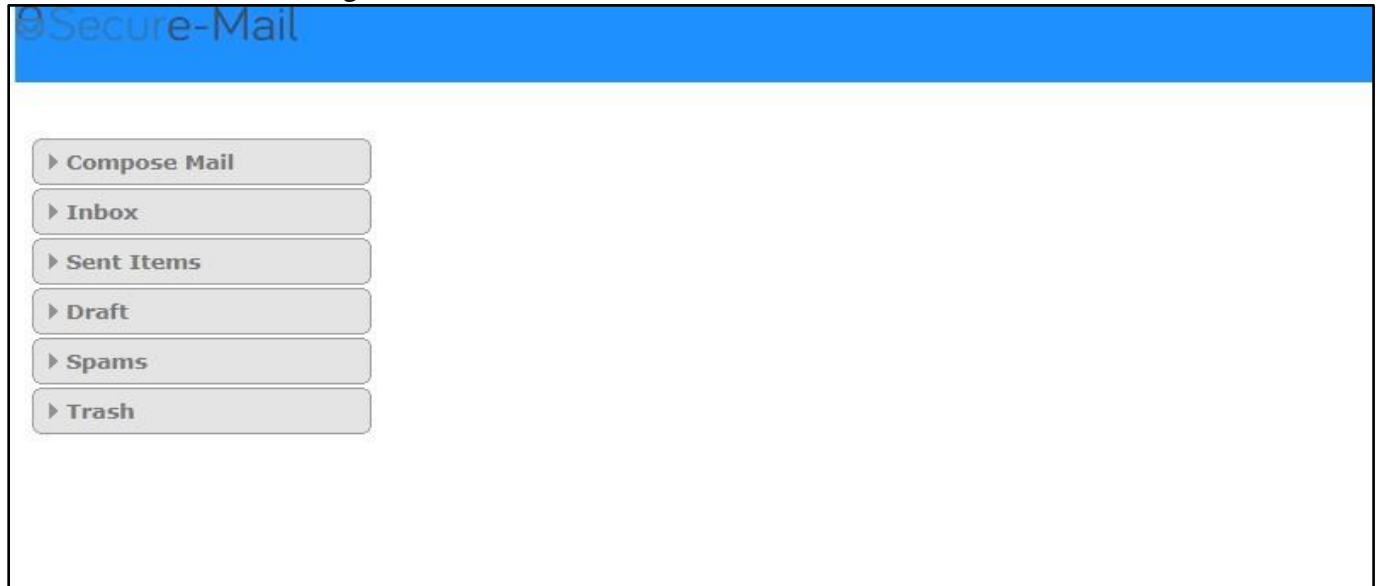

Fig. 4.7 User Homepage

4.4.8 This is the compose module where user can compose and send mails to other users who are already registered on the website.

Fig. 4.8 Compose Page

4.4.9 This is the inbox of user where user can check the received mails from the other users.

Fig. 4.9 User Inbox

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENTS

5.1 Conclusion

Email has been a gainful and well known correspondence instrument because of amount of online customer's extension. In various communication and security applications through which information is exchanged, it is essential to distinguish flabbergasting correspondence in email. So, in the view of that in this system the sends are designated suspicious or common by detecting the keywords and mixed watchwords. The sends consists of these watchwords and encoded catchphrases are appointed suspicious sends and checked and blocked by the authorities. The proposed work will be valuable for diminishing the crimes by recognizing the suspicious email and will assistance the authorities to gather the data so as to take convincing actions against them.

5.2 Future Enhancements

There is continually scope for further improvement of the application despite of the manner the present application satisfies the necessities of the application required. As indicated by the creating changes and new structures, further work should be conceivable to upgrade the application. Since endeavor is organized in a versatile programming.

This application can be executed on web by purchasing system space and making a webpage.

The backend of this application uses MySQL database as it requires a large database.

As we use mixed watchword in this system, we could execute many encoded calculations like DES/3DES, RSA, AES, BLOWFISH, IDEA and some more.

References

[1] Sarwat Nizamani, Nasrullah Memon, Uffe Kock Wiil, Panagiotis Karampelas, "Modeling Suspicious Email Detection using Enhanced Feature Selection", 2013

[2] Samruddhi Rane, Gargi Moholkar , Kajal Yerunkar , Prof. Nilima Nikam, "The Detection of Suspicious Email Based on Decision Tree", 3, Mar-2017

[3] Nisha Rani, Mrs. Neetu Sharma, "The Suspicious E-Mail Detection via Triple DES Algorithm: Cryptography Approach", 5, May-2015, pg. 552-565

[4] S. Appavu alias Balamurugan and Ramasamy Rajaram, "The Suspicious E-Mail Detection via Decision Tree: A data mining approach", 2017, pg. 161-169

[5] S. Appavu, R. Rajaram, "Association rule mining for suspicious email detection: a data mining approach", *In Proc. of the IEEE International Conference on Intelligence and Security Informatics*, New Jersey, USA, 2007, pp. 316-323.

[6] S. Appavu and R. Rajaram, (2008). Learning to classify threatening e-mail. *Int. Journal of Artificial Intelligence and Soft Computing*. 1(1). pp. 39-51.

Books Referred

1. JAVA TECHNOLOGIES
    1.1 JAVA Server Pages by Larne Pekowsley
    1.2 JAVA Server Pages by Nick Todd

2. JDBC
    2.1 JAVA Database Programming with JDBC by Patel Moss

3. SOFTWARE ENGINEERING
    3.1 Software Engineering by Roger Pressman

4. HTML

    4.1 Holzner, HTML Black Book

    4.2 Larne Pekowsley, JAVA Server Pages