Web Phishing Detection

NALAIYA THIRAN PROJECT BASED ON LEARNING PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

Project Report

Team ID - PNT2022TMID36550
Amysoj J A Exson Joseph (TL)
Manish Kumar Jha
Sai Madan Raj TM
Sudeep S

Contents

1. INTRODUCTION

1.1 Project Overview

One of the most serious cyberattacks for which researchers are looking for a fix is phishing. In phishing, criminals seduce end users to obtain their private information. Phishing must be identified as soon as feasible in order to reduce the harm it does. Spear phishing, whaling, vishing, smishing, pharming, and other phishing attacks are only a few examples. There are several methods for detecting phishing, including whitelisting, blacklisting, content- and URL-based methods, visual similarity methods, and machine-learning methods. Various phishing attacks, attack channels, and approaches for identifying phishing sites are covered in this study. There is a performance comparison of nine different dataset sources and 18 different models. A list of difficulties with phishing detection methods is also provided..There are a number of users who purchase products online and make payment through various websites.There are multiple websites who ask users to provide sensitive data such as username, password or credit card details etc. often for malicious reasons. This type of website is known as a phishing website.The phishing website can be detected based on some important characteristics like URL and Domain Identity, and security and encryption criteria in the final phishing detection rate.

1.2 Purpose

The main purpose of the project is to detect the fake or phishing websites who are trying to get access to the sensitive data or by creating the fake websites and trying to get access of the user personal credentials.We are using machine learning algorithms to safeguard the sensitive data and to detect the phishing websites who are trying to gain access on sensitive data

2. LITERATURE SURVEY

2.1 Existing Problem

Phishing offenses are on the rise, costing billions of dollars. In these attacks, users enter sensitive information into a bogus website that appears to be legitimate. The most common phishing targets are SaaS and webmail sites. The phisher creates websites that resemble the benign websites. The phishing website link is then distributed to millions of internet users via email and other forms of communication. Emails, instant messages, or phone calls are commonly used to initiate these types of cyber-attacks. The goal of a phishing attack is not only to steal the victims' identities, but it can also be used to spread other types of malware such as ransomware, exploit approach weaknesses, or profit financially. The number of phishing attacks has increased since March, according to the Anti-Phishing Working Group (APWG) report for the third quarter of 2020, with 28,093 unique phishing sites detected between July and September. In the third quarter, the average amount demanded during wire transfer Business E-mail Compromise (BEC) attacks was $48,000, down from $80,000 in the second quarter and $54,000 in the first.

2.2 Reference

- Yu WD, Nargundkar S, Tiruthani N (2008) A phishing vulnerability analysis of web-based systems. Link: https://bit.ly/2VJhDer
- Sheng S, Holbrook M, Kumaraguru P, Cranor LF, Downs J (2010) Who falls for phish? a demographic analysis of phishing susceptibility and effectiveness of interventions.Link: https://bit.ly/2VL0NeA
- Sheng S, Wardman B, Warner G, Cranor LF, Hong J, et al. (2009) An empirical analysis of phishing blacklists. Link: https://bit.ly/3Az9TdT

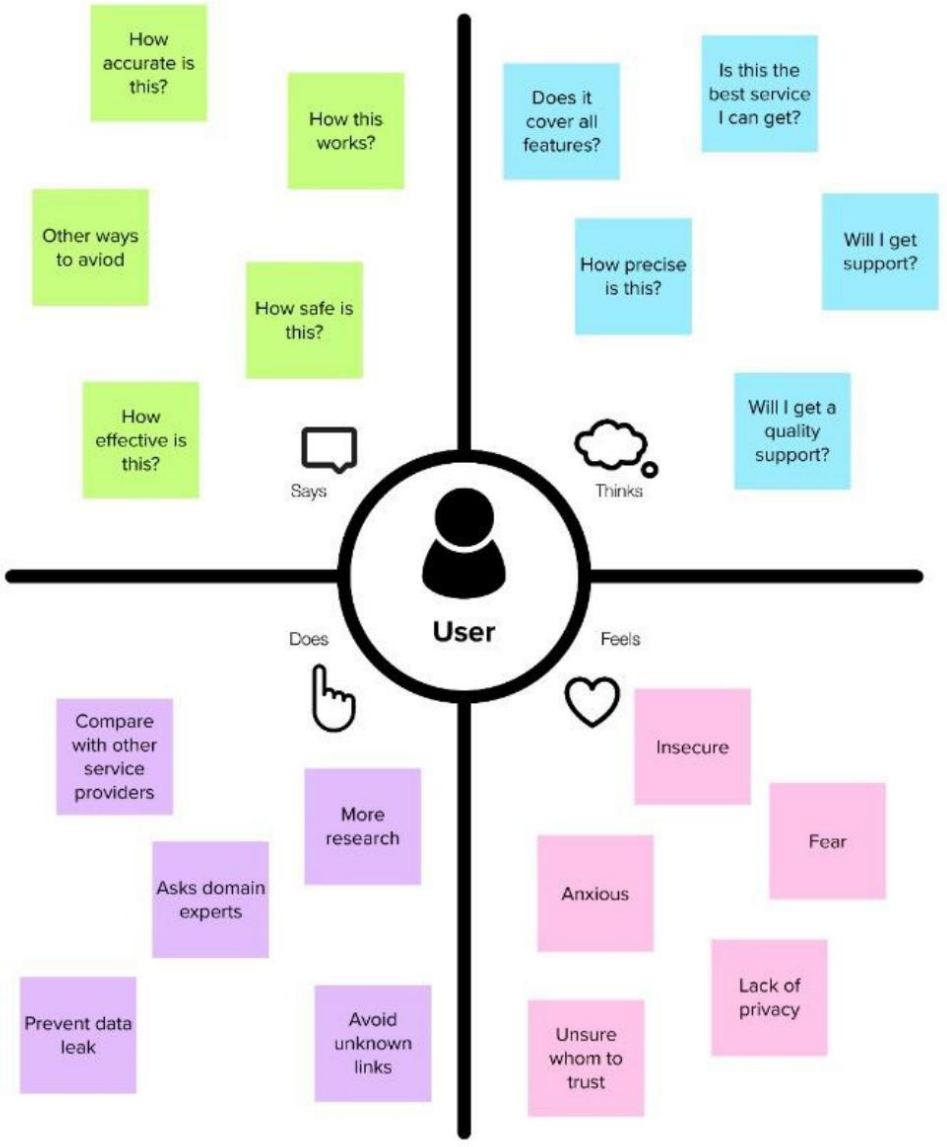
2.3 Problem Statement Definition

We can protect the user by URL detection and other simple techniques and identify the problem. Malicious links will lead to a website that often steals login credentials or financial information like credit card numbers. Roughly 214,345 unique phishing websites were identified, and the number of recent phishing attacks has doubled since early 2020. To identify the phishing websites a ML model to be created and analyze whether it is a legit website or a phishing trap so that users can be alerted.Phishing detection techniques do suffer low detection accuracy and high false alarm especially when novel phishing approaches are introduced. Besides, the most common technique used, blacklist-based method, is inefficient in responding to emanating phishing attacks since registering a new domain has become easier, no comprehensive blacklist can ensure a perfect up-to-date database. Furthermore, page content inspection has been used by some strategies to overcome the false negative problems and complement the vulnerabilities of the stale lists. Moreover, page content inspection algorithms each have different approaches to phishing website detection with varying degrees of

accuracy. Therefore, ensemble can be seen to be a better solution as it can combine the similarity in accuracy and different error-detection rate properties in selected algorithms. The user must be protected from the phishing attacks through similarly looking fake websites which look like exact clones of the original website that the phisher is trapping for users.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

● Build an AI powered phishing detection model that can detect whether it is a genuine website or an identical fake site.

● The ML model will learn the possible ways of the phishing website that can lure a user to get phished and the phisher gets away with it.

● This kind of phishing is prevented by the model and alerts the user about the activity.

● Build Good looking frontend.

● Analyzing the characteristics of the URL is another method of phishing detection. For instance, occasionally a URL resembles the URL of a well-known website or contains unusual letters.

● Make the UI user friendly and not to over-engineer and make things complicated for the user.

● Educate the user about the working of the application.

● Encourage people to use only https URL.

● We can fork another project and optimize it.

● We can include secure routing for higher security.

● Try to do something different from research problems in wireless networks.

● Using a white list or black list is the most straightforward technique to determine whether a particular website is engaging in web phishing.

● The Backpropagation is a supervised way.

● Promote use of many well-known browser vendors such as Firefox and Chrome.

3.3 Proposed Solution

- **Problem Statement (Problem to be solved)**

  The increase in the number of online phishing dramatically over the years have led the spark to build the phishing detection.

- **Idea / Solution description**

  To secure the users from phishers, to avoid the loss of personal details like geo-location, banking credentials etc through various kinds of engineered tools to detect the unusual activities in the user's machine and alert them.

- **Novelty / Uniqueness**

  Our application is very easy to use with a simple UI and very user friendly which is well designed for the intended purpose and very light-weight which you can not even compare with other similar services.

- **Social Impact / Customer Satisfaction**

  It'll create awareness among users to be secure online. The users will be very satisfied that they are in safe hands and need not worry about the problems described earlier.

- **Business Model (Revenue Model)**

  Revenue will be generated through charging a price(subscription) to the advanced features and basic features will be free of cost.

- **Scalability of the Solution**

  By securing the user base and earning their trust securing enterprises and scaling the business and capturing the market-share.

3.4 Problem Solution Fit

1. CUSTOMER SEGMENT

Our customers are those who uses a computer and needs a solution to protect their privacy in the online jungle.

6. CUSTOMER CONSTRAINTS

Not even a single thing constraints our customer from using the product because our product is a free to use and we only charge for the premium features.

5. AVAILABLE SOLUTIONS

They might have used some kind of anti-virus software that only checks the https connection. We check the weather the spelling of the website is correct or not through a database of most visited websites of a day-to-day user.

2. JOBS-TO-BE-DONE / PROBLEMS

Phishing is one of the main concerns these days and the number of phishing attacks have increased over the years. We are focusing on preventing phishing websites from phishing.

9. PROBLEM ROOT CAUSE

There are mirror websites similar to the legit one which people might find difficult to find legit one.

Here we come in and check weather the website is legit one or not.

7. BEHAVIOUR

User uses the product, which makes an impact in their life.

Spreads the product to their friends and family which they use to in their day to day life and how it is useful to them.

3. TRIGGERS

When someone got phished (weather its their neighbours or someone in the news) and lost their money in the bank account.

4. EMOTIONS: BEFORE / AFTER

The customers feel very safe and confident after using the product.

10. YOUR SOLUTION

Most of the users are not aware of the possibility of getting phished even through simple means.

Our solution for this problem is to create awareness among the users and provide incentives to those who reach a certain amount of (may be) points. The higher the points they can get free premium subscription. By this way money might not be the constraint.

8. CHANNELS of BEHAVIOUR

8.1 ONLINE

Users should not click any links that they never heard of or suspicious or even misspelled(sometimes which look similar to the legit one).

8.2 OFFLINE

User needs to scan the computer to check weather phisher are monitoring them through backdoor.

4. REQUIREMENT ANALYSIS

Functional Requirements

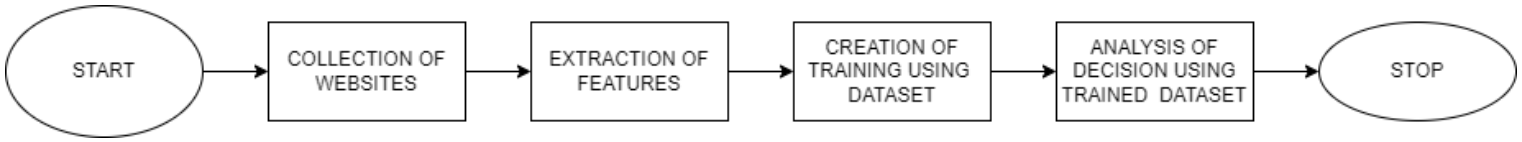
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Registration**	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	**User Confirmation**	Confirmation via Email Confirmation via OTP
FR-3	**Welcoming User**	Welcome user with a welcome page
FR-4	**User Purchase**	Redirect to the payment gateway
FR-5	**User Purchase confirmation**	Confirmation via Email Confirmation via OTP
FR-6	**Features**	Showing the features after the purchase

Non-functional Requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It's very easy to use the application in terms of complexity and very user friendly.
NFR-2	**Security**	The main purpose of this very project is security and it will not be compromised under any circumstances.
NFR-3	**Reliability**	Security is our main goal and our vision, securing our users is our work. Users can be very reliable on the product.
NFR-4	**Performance**	The application will not use the most of the CPU performance instead it will use the data hosted in the cloud.
NFR-5	**Availability**	Anybody having a PC with internet connection can use the available services.
NFR-6	**Scalability**	Scaling through creating awareness among the users and providing incentives to those who invite new users.

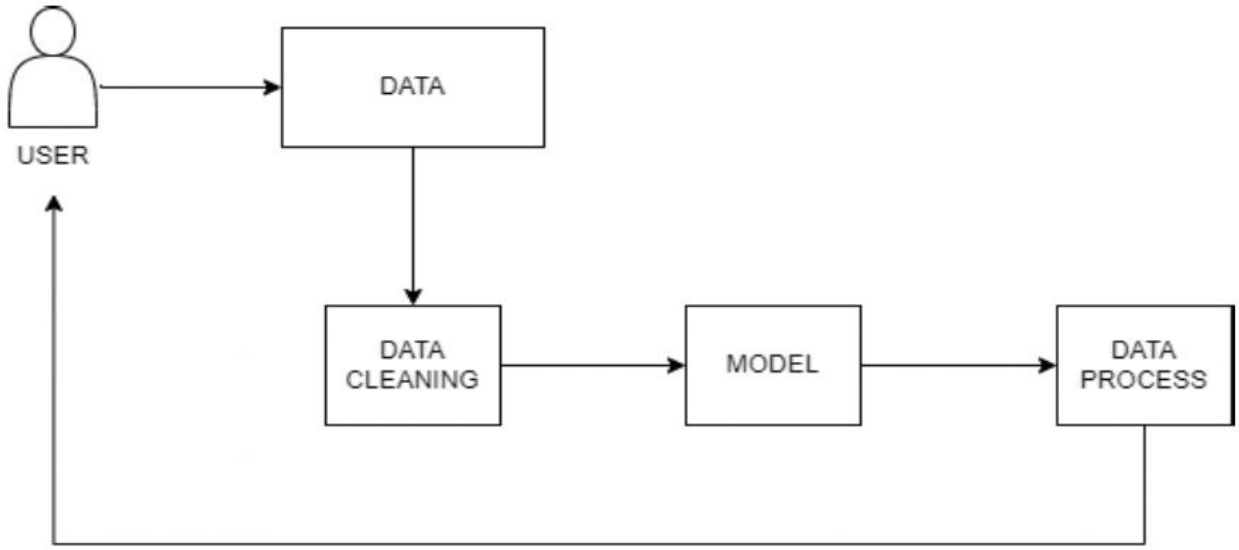
5. PROJECT DESIGN

5.1 Data Flow Diagram

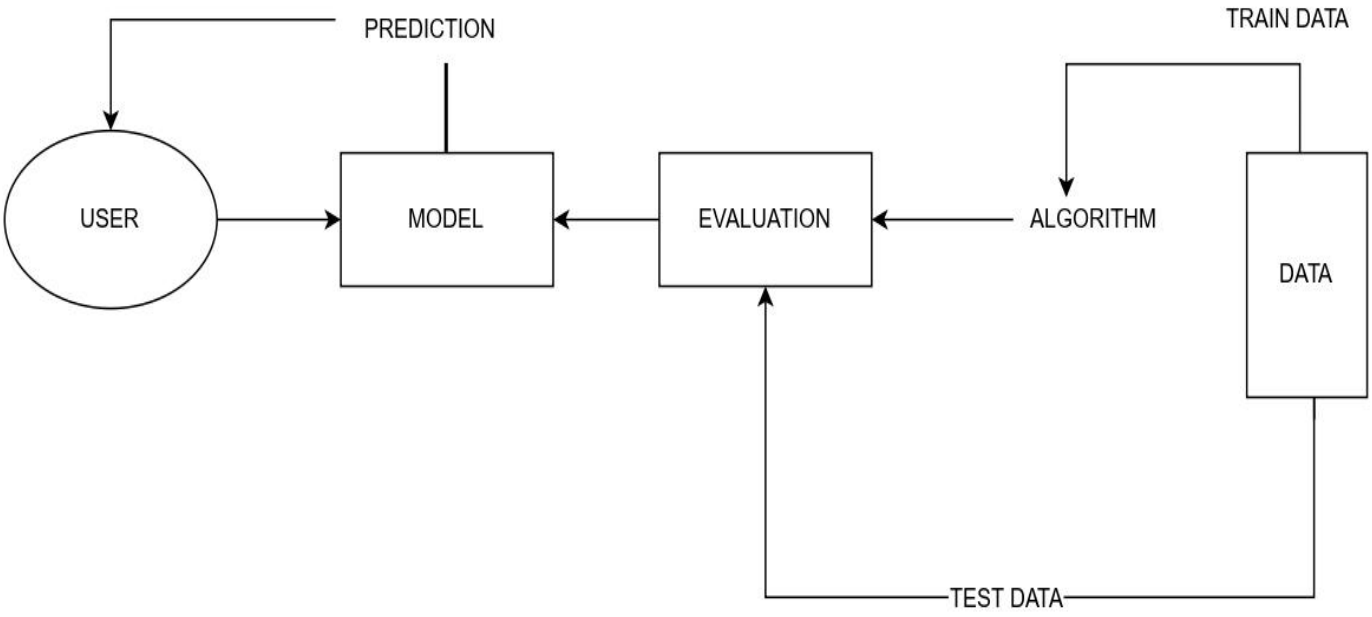


5.2 Solution & Technical Architecture

=> Solution Architecture



=> Technical Architecture



5.3 User Stories

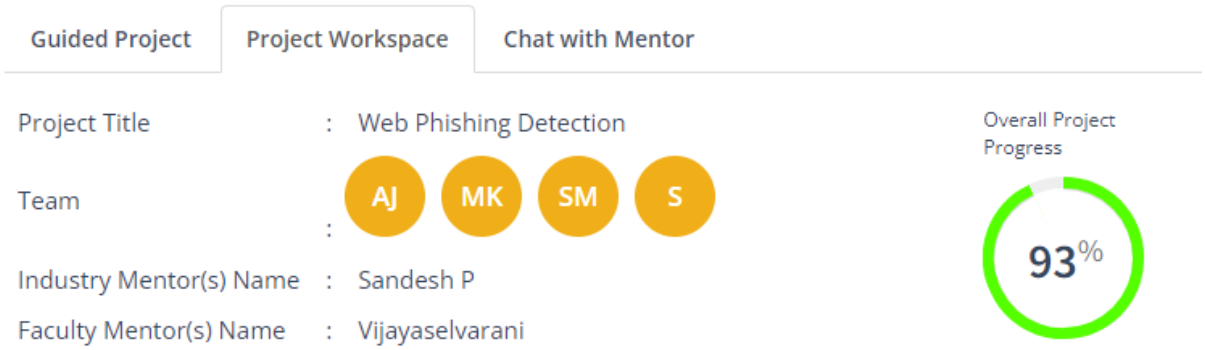
User 1	As a user, I can explore the resources of the homepage for the functioning.
User 2	As a user, I can explore the resources of the final page for the functioning.
User 3	As a user, I can predict the URL easily for detecting whether the website is legitimate or not.
User 4	As a user, I can share the experience or contact the admin for the support.
User 5	As an admin, we can design interfaces and maintain the functioning of the website.
User 6	As an admin, we can design the complexity of the website to make it user-friendly.
User 7	As an admin, we can use various ML classifier models for the accurate result for the detection of URL.
User 8	As an admin, we can respond to the user message for improvement of the website.

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning, Estimation and Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Planned Date	Sprint Estimated Date	Story Points Completed	Sprint Delivery Date
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	12 Nov 2022

6.2 JIRA Report



7. CODING AND SOLUTIONS

Source Code (Logic)

```
import regex
from tldextract import extract
import ssl
import socket
from bs4 import BeautifulSoup
import urllib.request
import whois
import datetime
import requests
import favicon
import re
from googlesearch import search

def having_IPhaving_IP_Address(url):
     match=regex.search(

'(([01]?\\d\\d?|2[0-4]\\d|25[0-5])\\.([01]?\\d\\d?|2[0-4]\\d|25[0-5])\\.([
01]?\\d\\d?|2[0-4]\\d|25[0-5])\\.([01]?\\d\\d?|2[0-4]\\d|25[0-5])\\/)|'
#IPv4

'((0x[0-9a-fA-F]{1,2})\\.(0x[0-9a-fA-F]{1,2})\\.(0x[0-9a-fA-F]{1,2})\\.(0x
[0-9a-fA-F]{1,2})\\/)'  #IPv4 in hexadecimal
                    '(?:[a-fA-F0-9]{1,4}:){7}[a-fA-F0-9]{1,4}',url)
#Ipv6
     if match:
        return -1
     else:
        return 1

def URLURL_Length (url):
    length=len(url)
    if(length<=75):
```

```
        if(length<54):
            return 1
        else:
            return 0
    else:
        return -1


def Shortining_Service (url):

match=regex.search('bit\.ly|goo\.gl|shorte\.st|go2l\.ink|x\.co|ow\.ly|t\.co|tinyurl|tr\.im|is\.gd|cli\.gs|'

'yfrog\.com|migre\.me|ff\.im|tiny\.cc|url4\.eu|twit\.ac|su\.pr|twurl\.nl|snipurl\.com|'

'short\.to|BudURL\.com|ping\.fm|post\.ly|Just\.as|bkite\.com|snipr\.com|fic\.kr|loopt\.us|'

'doiop\.com|short\.ie|kl\.am|wp\.me|rubyurl\.com|om\.ly|to\.ly|bit\.do|t\.co|lnkd\.in|'

'db\.tt|qr\.ae|adf\.ly|goo\.gl|bitly\.com|cur\.lv|tinyurl\.com|ow\.ly|bit\.ly|ity\.im|'

'q\.gs|is\.gd|po\.st|bc\.vc|twitthis\.com|u\.to|j\.mp|buzurl\.com|cutt\.us|u\.bb|yourls\.org|'

'x\.co|prettylinkpro\.com|scrnch\.me|filoops\.info|vzturl\.com|qr\.net|1url\.com|tweez\.me|v\.gd|tr\.im|link\.zip\.net',url)
    if match:
        return -1
    else:
        return 1

def having_At_Symbol(url):
    symbol=regex.findall(r'@',url)
    if(len(symbol)==0):
        return 1
    else:
```

```
        return -1

def double_slash_redirecting(url):
    for i in range(8,len(url)):
        if(url[i]=='/'):

            if(url[i-1]=='/'):
                return -1
    return 1

def Prefix_Suffix(url):
    subDomain, domain, suffix = extract(url)
    if(domain.count('-')):
        return -1
    else:
        return  1

def having_Sub_Domain(url):
    subDomain, domain, suffix = extract(url)
    if(subDomain.count('.')<=2):
        if(subDomain.count('.')<=1):
            return 1
        else:
            return 0
    else:
        return -1

def SSLfinal_State(url):
    try:
        response = requests.get(url)
        return 1
    except Exception as e:
        return -1

def Statistical_report (url):
    hostname = url
    h = [(x.start(0), x.end(0)) for x in
regex.finditer('https://|http://|www.|https://www.|http://www.',
hostname)]
    z = int(len(h))
```

```
    if z != 0:
        y = h[0][1]
        hostname = hostname[y:]
        h = [(x.start(0), x.end(0)) for x in regex.finditer('/',
hostname)]
        z = int(len(h))
        if z != 0:
            hostname = hostname[:h[0][0]]

url_match=regex.search('at\.ua|usa\.cc|baltazarpresentes\.com\.br|pe\.hu|e
sy\.es|hol\.es|sweddy\.com|myjino\.ru|96\.lt|ow\.ly',url)
    try:
        ip_address = socket.gethostbyname(hostname)

ip_match=regex.search('146\.112\.61\.108|213\.174\.157\.151|121\.50\.168\.
88|192\.185\.217\.116|78\.46\.211\.158|181\.174\.165\.13|46\.242\.145\.103
|121\.50\.168\.40|83\.125\.22\.219|46\.242\.145\.98|107\.151\.148\.44|107\
.151\.148\.107|64\.70\.19\.203|199\.184\.144\.27|107\.151\.148\.108|107\.1
51\.148\.109|119\.28\.52\.61|54\.83\.43\.69|52\.69\.166\.231|216\.58\.192\
.225|118\.184\.25\.86|67\.208\.74\.71|23\.253\.126\.58|104\.239\.157\.210|
175\.126\.123\.219|141\.8\.224\.221|10\.10\.10\.10|43\.229\.108\.32|103\.2
32\.215\.140|69\.172\.201\.153|216\.218\.185\.162|54\.225\.104\.146|103\.2
43\.24\.98|199\.59\.243\.120|31\.170\.160\.61|213\.19\.128\.77|62\.113\.22
6\.131|208\.100\.26\.234|195\.16\.127\.102|195\.16\.127\.157|34\.196\.13\.
28|103\.224\.212\.222|172\.217\.4\.225|54\.72\.9\.51|192\.64\.147\.141|198
\.200\.56\.183|23\.253\.164\.103|52\.48\.191\.26|52\.214\.197\.72|87\.98\.
255\.18|209\.99\.17\.27|216\.38\.62\.18|104\.130\.124\.96|47\.89\.58\.141|
78\.46\.211\.158|54\.86\.225\.156|54\.82\.156\.19|37\.157\.192\.102|204\.1
1\.56\.48|110\.34\.231\.42',ip_address)
    except:
        return -1

    if url_match:
        return -1
    else:
        return 1

def main(url):
```

```
    check = [[having_IPhaving_IP_Address
(url),URLURL_Length(url),Shortining_Service(url),having_At_Symbol(url),

double_slash_redirecting(url),Prefix_Suffix(url),having_Sub_Domain(url),SS
Lfinal_State(url),

Domain_registeration_length(url),Favicon(url),port(url),HTTPS_token(url),R
equest_URL(url),

URL_of_Anchor(url),Links_in_tags(url),SFH(url),Submitting_to_email(url),Ab
normal_URL(url),

Redirect(url),on_mouseover(url),RightClick(url),popUpWidnow(url),Iframe(ur
l),

age_of_domain(url),DNSRecord(url),web_traffic(url),Page_Rank(url),Google_I
ndex(url),
              Links_pointing_to_page(url),Statistical_report(url)]]


    print(check)
    return check
```

8. TESTING

User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

3. Test Case Analysis

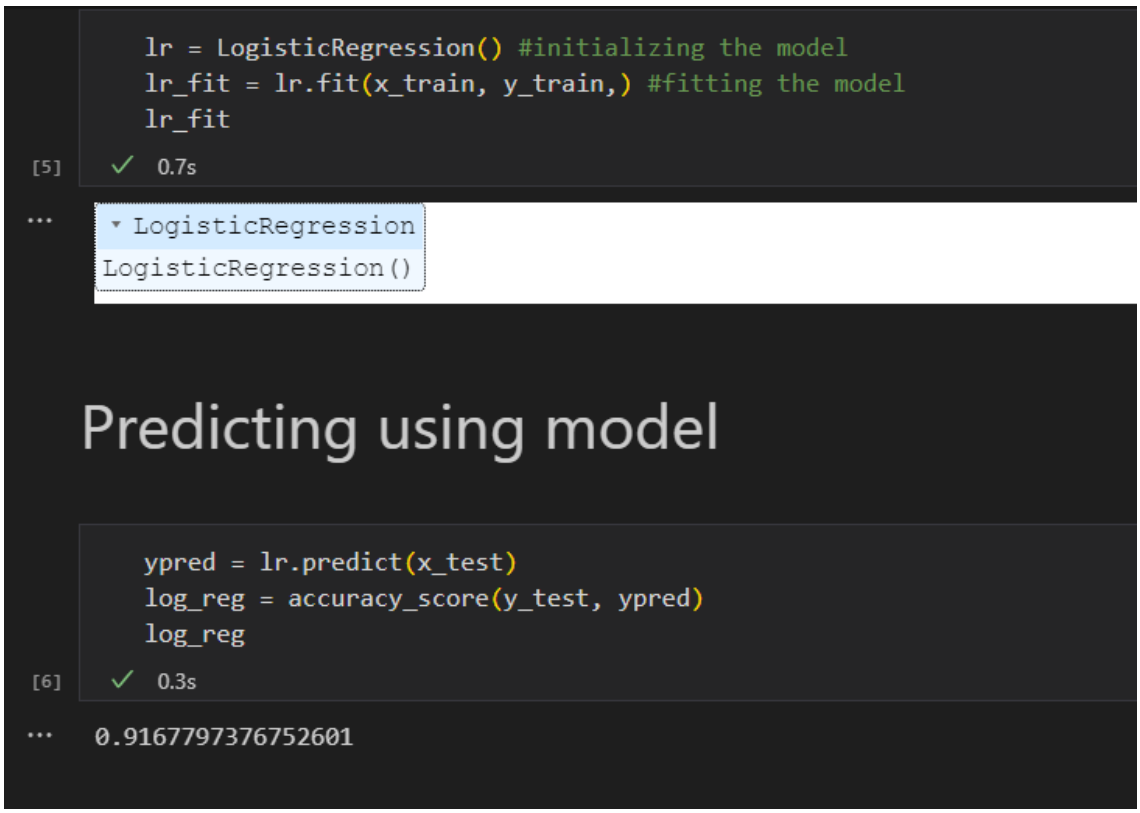
This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

Sl No	Parameters	Values
1	Model Accuracy	Accuracy : 0.91
2	Metrics	Regression Model : Logistic Regression MAE : 0.16 MSE : 0.33 RMSE : 0.57 R2 Score : 0.66

Model Accuracy



Metrics

Evaluation Metrics

```
MAE = mean_absolute_error(ypred, y_test)
MAE
```

[15] ✓ 0.2s

```
0.16644052464947987
```

```
MSE = mean_squared_error(ypred, y_test)
MSE
```

[16] ✓ 0.3s

```
0.33288104929895973
```

```
RMSE = np.sqrt(MSE)
print(RMSE)
```

[21] ✓ 0.3s

```
0.5769584467697476
```

```
r2_score = r2_score(ypred, y_test)
r2_score
```

[37] ✓ 0.7s

```
0.6627729239543096
```

10. ADVANTAGES AND DISADVANTAGES

Advantages

- Improves Phishing Awareness Training Inefficiencies
- It Provides a Solution Rather Than a Tool
- Set You Apart from Your Rivals
- Many e-commerce companies can use this technique to maintain positive consumer interactions.
- This system will function even without an internet connection.

Disadvantages

- New data should be added manually.
- Data Entry is a time consuming process.
- Data pertaining to the website will be kept in one location.
- It takes a long time to complete the process.

11. CONCLUSION

Despite advancements in intelligence and security, phishing is still on the rise, thus special attention must be paid to protecting those who have been defrauded. In this study, the performance of various machine learning algorithms on a phishing dataset was examined. It was discovered that random forest performs better in terms of accuracy, error rate, and other factors. The proposed approach has also been compared to other, comparable works, and it has been discovered that, when compared to works presented by different authors, the proposed model achieves significantly superior accuracy. The results reveal that the suggested model achieves much improved accuracy as compared to works reported by other authors. The proposed algorithm has also been compared with work on different datasets using comparable algorithms. Nowadays, criminals are using phishing as a highly profitable activity. Over the past several years, there has been an increase in the technology, diversity, and sophistication of these attacks in response to increased user awareness and countermeasures, in order to maintain profitability.Phishing differs from traditional scams primarily in the scale of the fraud that can be committed.The problem of Phishing does not have a single solution as of today. Phishing is not just a technical problem and Phishers would keep coming up with new ways of attacking the users. Online users should undertake periodic vulnerability analysis to identify and plug weaknesses that can lead to a successful Phishing attack.

12. FUTURE SCOPE

In the future, phishing detection will be considerably faster than with any other technique if we have access to structured datasets of phishing. In the future, we can combine any other two or more classifiers to achieve the highest accuracy. We also intend to investigate other phishing methods that take advantage of lexical, network, and content-based aspects to enhance system efficiency. We specifically extract information from URLs and subject them to different classifiers. We can categorize data into phishing, suspect, and valid with the aid of machine learning techniques like Random Forest, Decision Tree, Neural Network, and Linear Model. The user does not need to inspect each website individually because this can be done based on the distinctive traits of phishing websites. Instead, by identifying and anticipating specific characteristics, we can distinguish between legal, suspect, and phishing websites. The development of a model to protect users from phishing attacks was the goal of this work. On a phishing dataset, methods from Random Forest, Decision Tree, Linear Model, and Neural Network will be applied. These algorithms' outputs will be contrasted in terms of recall, precision, recall rate, accuracy, and error rate. As a future work on phishing, it must be a better trend to work on server side security. In the server side security policy , a dual level of authentication can be used for a user by which only authentic users can get the access of his account, and to educate the user about this policy will result in avoiding the user giving his sensitive information to a phished website.

13. APPENDIX

GitHub & Project Demo Link

Github
https://github.com/IBM-EPBL/IBM-Project-43716-1660718978

Project Demo
https://python-flask-app-yipwg-2022-11-23-talkative-antelope-vz.mybluemix.net/