KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

HX 8001-PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

WEB PHISHING DETECTION

NALAIYA THIRAN PROJECT REPORT 2022

Submitted by

DHARUN S	**621319104013**
ABINASH B	**621319104001**
KISHORE G	**621319104026**
VASANTHKUMAR D	**621319104062**

Team ID : PNT2022TMID13259

Industry mentor : **Sandesh P**

Faculty mentor : **Lalitha K**

NOVEMBER 2022

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1. PROJECT OVERVIEW

Due of how simple it is to develop a phoney website that closely resembles a legitimate website, phishing is now a top worry for security researchers. Although experts can spot fraudulent websites, not all users can, and as a result, some users fall prey to phishing scams. The attacker's primary goal is to obtain login information for bank accounts. This is the standard technique to identify phishing websites Internet Protocol (IP) addresses that have been blacklisted are added to the antivirus database using the "blacklist" technique. to avoid Attackers on blacklists craftily alter the URL to appear legitimate by obfuscation and many other straightforward ways, such as fast-flux, in which proxies are automatically built to host the website.

1.2. PURPOSE OF THE PROJECT

Phishing attacks are the simplest way to obtain sensitive information from unaware individuals. Phishers seek to get sensitive information such as usernames, passwords, and specifications of bank accounts. Professionals in cyber security are currently searching for effective and dependable ways to identify phishing websites. By extracting and contrasting a number of factors, this study investigates the application of machine learning algorithms to discriminate between legitimate and phishing URLs. The algorithms Decision Tree, Random Forest, and Support Vector Machine are used to detect phishing websites. The goal of the project is to identify phishing URLs and determine the most effective machine learning technique by comparing the accuracy rates, false positive, and false negative rates of each algorithm.

CHAPTER 2

LITERATURE SURVEY

[1] TITLE : Smart Phishing Detection in Web Pages.

Authors: Chukka Santhaiah & U. Janardhan Reddy / 2021

Proposed Problem:

Bayesian classification is used in current phishing detection methods to distinguish between malicious and legitimate web pages.

Proposed solution:

To distinguish between phishing websites and legitimate websites, the suggested deep learning model with Adam Optimizer employs a List wise method.

Pros and cons:

When compared to other conventional machine learning algorithms like SVM, Adaboost, and AdaRank, the suggested approach performs reasonably well.

[2] TITLE: Towards Lightweight URL-Based Phishing Detection

Authors: Andrei Butnaru ,Alexios Mylonas/ 2021

Proposed Problem:

Phishing attacks typically act as an attack vector or the first step in a more complicated campaign in the current threat landscape.

Proposed solution:

This performance over time was determined using a dataset of active phishing attacks, and it was compared to Google Safe Browsing.

Pros and cons:

In every experiment, the work outperforms GSB, and it also works effectively against phishing URLs that are still live a year after our model was trained.

[3] TITLE: A Deep Learning Technique for Web Phishing Detection Combined URL Features and Visual Similarity

Authors: Saad Al Ahmadi / 2020

Proposed Problem:

It addresses the problem of phishing website recognition as a classification task by describing the website's appearance and URL.

Proposed solution:

The CNN was completely trained using this way. A data set made up of 129,450 clinical pictures is used to train CNN.

Pros and cons:

The model employs CNNs to identify a phishing attack using web page URLs and images. It identifies the key elements of website images and URLs before classifying them as phishing pages.

[4] TITLE:Meta Algorithms for improving Classification Performance in the Web phishing Detection Process

Authors: Anggit Ferdita Nugraha/ 2019.

Proposed Problem:

Many researchers have studied web phishing detection systems; one of them used data mining methods but still relied on a single classification algorithm.

Proposed solution:

To support the enhancement of classification Performance for the development of various web phishing detection systems, the addition of meta algorithm is proposed.

Pros and cons:

In the testing phase, using the Web Phishing dataset from the UCI Machine Learning Repository, the addition of the bagging process, the boosting process, and the stacking process resulted in accuracy increases of 97.1%, 97.3%, and 97.5%, respectively.

[5] TITLE: Web Phishing Detection Using a Deep Learning Framework

Authors: Yuxiang Guan,Futai Zou,Yao Yao,Wei Wang,and Ting Zhu/ 2018

Proposed Problem:

By seeming to be a trustworthy organisation, web phishing seeks to obtain sensitive information including usernames, passwords, and credit card numbers.

Proposed solution:

Original features and interaction features are two different categories of features for web phishing. The Deep Belief Networks-based detection model.

Pros and cons:

The experiment demonstrates that the suggested DBN-based detection model can achieve a true positive rate of about 90% and a false positive rate of about 0.6%.

[6] TITLE:Systematization of knowledge (SoK): A systematic review of software-based web phishing detection.

Authors: Z. Dou, I. Khalil, A. Khreishah, A. Al-Fuqaha and M. Guizani-2017

Proposed Problem:

Phishing is a type of cyberattack that uses sophisticated social engineering methods and other techniques to gather personal data from website visitors.

Proposed solution:

On the basis of their distinctive content, network, and URL characteristics, phishing attempts have been the subject of extensive research and development. Intuitions, data analysis techniques, and evaluation methodologies vary widely between existing approaches.

Pros and cons:

It is possible to compare and contrast each approach's benefits and drawbacks as well as its suitability for use in various situations.

[7] TITLE:Phishing detection: A recent intelligent machine learning comparison based on models content and features

Authors: Neda Abdelhamid, Fadi Thabtah, Hussein Abdel-jaber-2017

Proposed Problem:

The applicability of machine learning (ML) techniques to phishing attack detection is examined in this article, along with their benefits and drawbacks. In particular, various ML techniques have been researched to identify the best options that can be used as anti-phishing tools.

Proposed solution:

The comparison aims to highlight the benefits and drawbacks of ML prediction models and to demonstrate how well they actually perform when it comes to phishing attempts. More significantly, we compare a variety of ML techniques experimentally using actual phishing datasets and various metrics.

Pros and cons:

When compared to traditional anti-phishing methods, such as awareness seminars, visualisation, and legal remedies, machine learning (ML), a prominent tool for data analysis, has recently demonstrated encouraging outcomes in countering phishing. This article explores the use of ML approaches to identify phishing attacks and lists their benefits and drawbacks.

[8]TITLE:A survey and classification of web phishing detection schemes

Authors: Gaurav Varshney,Manoj Misra,Pradeep K. Atrey-2016

Proposed Problem:

Phishing is a dishonest method of tricking users online in order to steal their personal data, including usernames, passwords, credit card numbers, and bank account numbers. Phishing relies on deceit to work. Phishing starts with email spoofing for deceptive communication, then moves on to spoofed websites to get the information it needs from the victims.

Proposed solution:

It is among the most serious cybercrimes that Internet users have to deal with. There is currently no comprehensive and precise method for preventing phishing attacks, though researchers are working on their detection, prevention, and education.

Pros and cons:

The report highlights benefits, disadvantages, and research gaps in the field of phishing website identification that can be addressed in further studies and advancements. The analysis presented in this research will assist academia and business in determining the most effective anti-phishing method.

[9] TITLE: Detecting visually similar Web pages: Application to phishing detection

Authors: Teh-Chung Chen,Scott Dick,James Miller-2010

Proposed Problem:

Method for comparing the visual similarities of two Web pages. The suggested method makes use of Gestalt theory and treats a Web page as a single, unbreakable unit. Our argument that Web pages should be treated as indivisible entities is supported by the idea of supersignals, which is a realisation of Gestalt principles.

Proposed solution:

Using algorithmic complexity theory, we objectify and directly compare these indivisible supersignals. We use the issue of identifying phishing scams to illustrate our method.

Pros and cons:

We show through a sizable, real-world case study that 1) our approach accurately detects similar Web pages and 2) it can tell the difference between legitimate and phishing pages.

[10] TITLE:An Automated Web Phishing Detection Approach Based on Profiling and Fuzzy Matching

Authors: Sadia Afroz and Rachel Greenstadt-2009

Proposed Problem:

Phishing is a web-based assault that makes use of social engineering methods to take advantage of Internet users and obtain private information.

Proposed solution:

In order to earn the user's trust, the majority of phishing attempts use a false web interface for the legitimate site. Despite the fact that some phishing sites closely resemble the legitimate websites they pretend to be, user studies have shown that users frequently disregard browser-based signs and assess a site's validity based solely on its appearance, just as they would an actual physical location.

Pros and cons:

By lessening the effectiveness of websites that resemble legitimate websites too closely and giving users a chance to identify websites that "look phishy," PhishZoo has the potential to have a positive effect on the phishing "arms race."

2.1. EXISTING PROBLEM

The issue with phishing is that attackers are always coming up with fresh and original techniques to trick victims into thinking their activities are connected to a reliable website or email. Phishers are getting better at creating fake websites that look exactly like the real thing. They've even started adding logos and pictures to their phishing emails to increase their effectiveness.

2.2. REFERENCES

1. S R Sharma, R Parthasarathy and P B Honnavalli, "A Feature Selection Comparative Study for Web Phishing Datasets[C]", 2020 IEEE International Conference on Electronics Com-puting and Communication Technologies (CONECCT), 2020.
2. A. Aljofey, Q. Jiang, Q. Qu, M. Huang and J.-P. Niyigena, "An effective phishing detection model based on character level convolutional neural network from URL", Electronics, vol. 9, no. 9, pp. 1514, Sep. 2020.
3. D Dua and C Graff, "UCI machine learning repository. University of California School of Information and Computer Science", Irvine CA[J], 2019
4. Y. Ding, N. Luktarhan, K. Li and W. Slamu, "A keyword-based combination approach for detecting phishing webpages", Comput. Secur., vol. 84, pp. 256-275, Jul. 2019.
5. Z. Dou, I. Khalil, A. Khreishah, A. Al-Fuqaha and M. Guizani, "Systematization of knowledge (SoK): A systematic review of software-based web phishing detection", IEEE Commun. Surveys Tuts., vol. 19, no. 4, pp. 2797-2819, 4th Quart. 2017.
6. F Thabtah, H Abdel-jaber, "Phishing detection: A recent intelligent machine learning comparison based on models content and features "- IEEE international …, 2017
7. S Afroz, R Greenstadt ,"An automated web phishing detection approach based on profiling and fuzzy matching"- Proc. 5th IEEE Int. Conf. Semantic Comput …, 2009

2.3. PROBLEM STATEMENT

Many consumers utilize e-banking to make purchases and payments for goods they find online. There are several e-banking websites that often request sensitive information from users—such as usernames, passwords, and credit card information—for harmful purposes. Phishing websites are this kind of online banking websites. One of the most important software services for communications on the Internet is the web service. One of the many security risks to online services on the Internet is web phishing. By seeming to be a trustworthy organization, web phishing seeks to obtain sensitive information including usernames, passwords, and credit card numbers. It will result in data leakage and property damage. Large businesses may fall victim to various schemes.

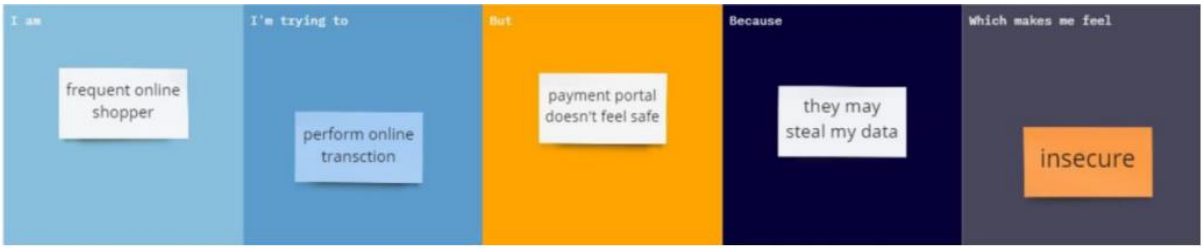


Fig. 2.2(a). Problem Statement

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Internet User	Browse the internet	I identify a scam	An attacker poses as a trustworthy organization.	Concerned about the security of my online information
PS-2	Enterprise user	Open emails in the cloud server	I detect malicious protocols	They are not cryptographic ally signed	Emails are unverified and third party intrusion

Fig. 2.2(b). Defining Problem Statement

CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1. EMPATHY MAP CANVAS

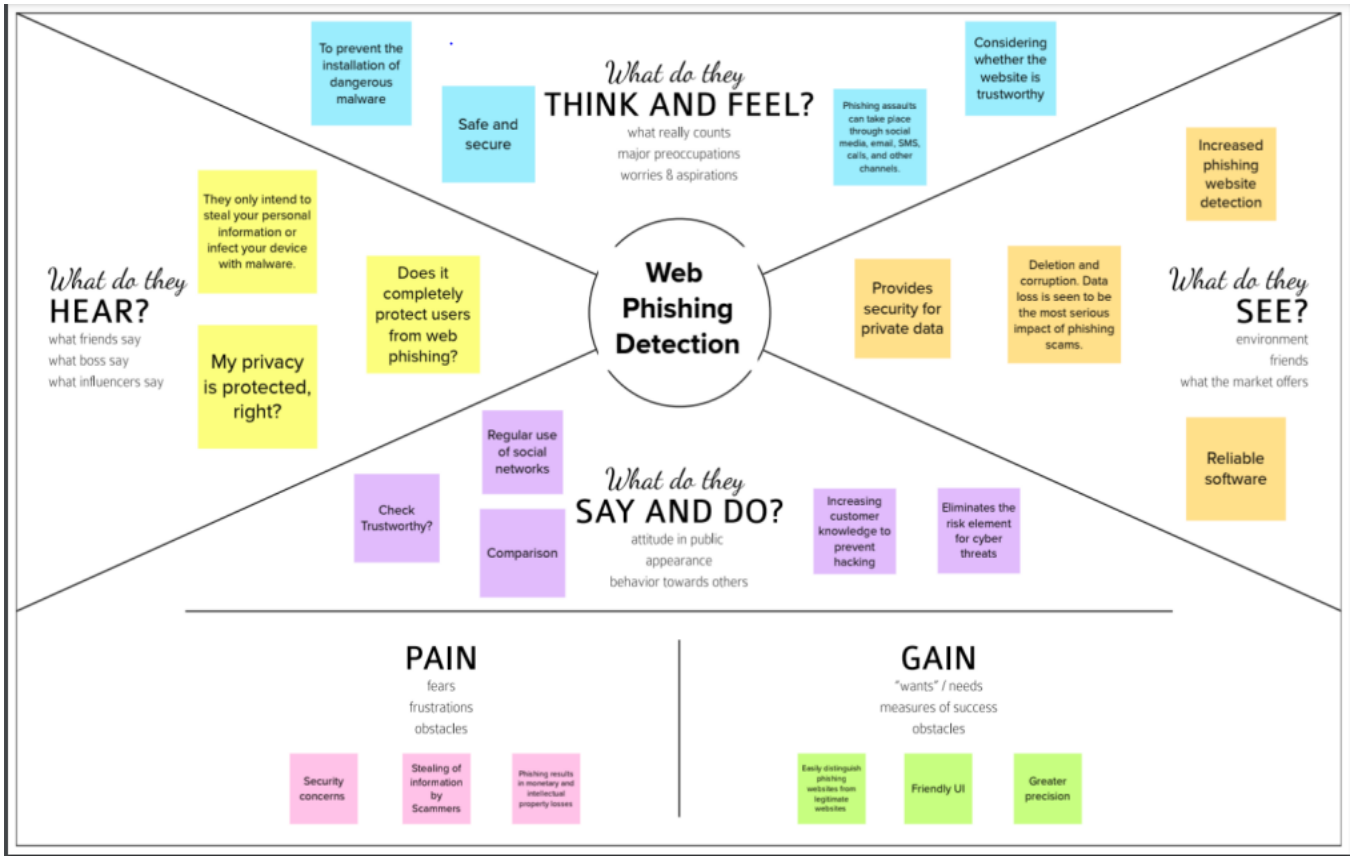


Fig. 3.1. Emphathy Map

3.2. IDEATION AND BRAIN STORMING

STEP 1: TEAM COLLABORATION & SELECTING THE PROBLEM STATEMENT

Fig. 3.2(a) Team Collaboration

Fig. 3.2(b) Selecting the problem statement

STEP 2: Brainstorm, Idea Listing and Grouping

Fig. 3.2(c) Brain storm

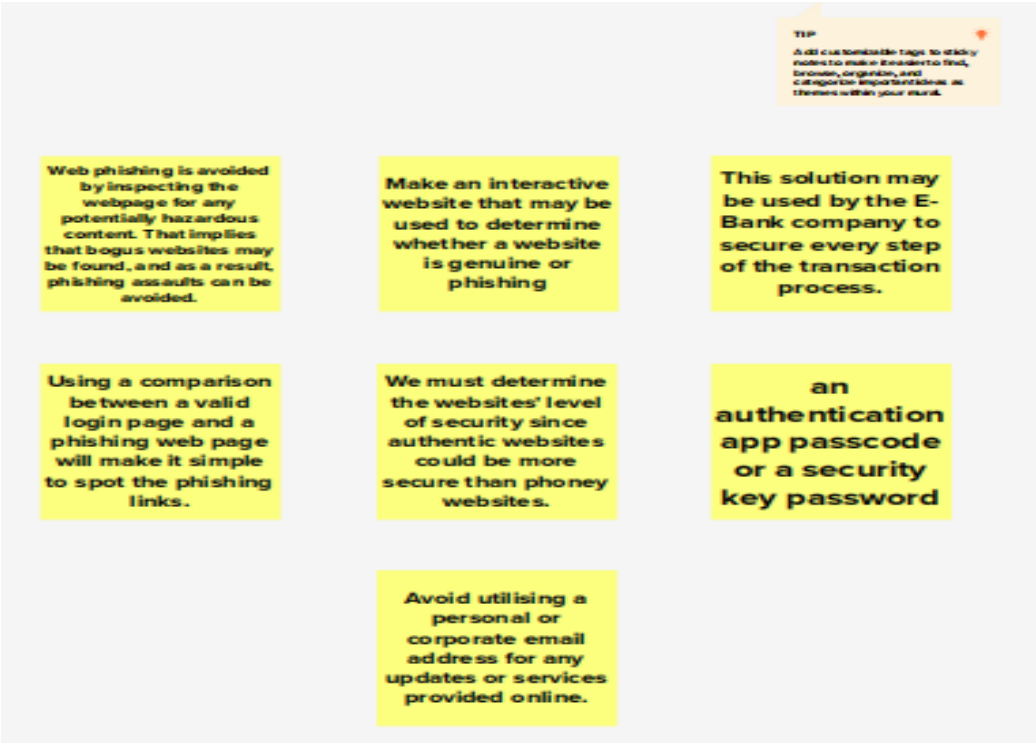


Fig. 3.2(d) Idea Listing and Grouping

STEP 3: IDEA PRIORITIZATION

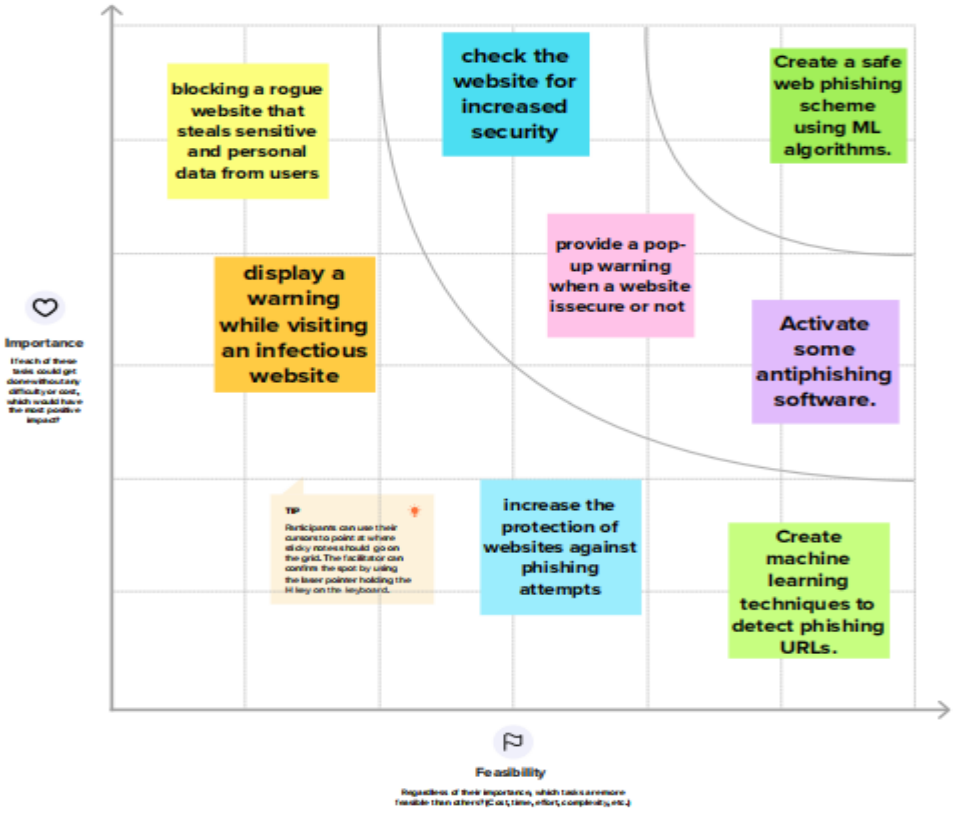


Fig. 3.2(e) Idea Prioritization

3.3. PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to besolved)	Many consumers utilize e-banking to make purchases and payments for goods they find online. There are several e-banking websites that often request sensitive information from users—such as usernames, passwords, and credit card information—for harmful purposes. Phishing websites are this kind of online banking websites. One of the most important software services for communications on the Internet is the web service. One of the many security risks to online services on the Internet is web phishing. By seeming to be a trustworthy organization, web phishing seeks to obtain sensitive information including usernames, passwords, and credit card numbers. It will result in data leakage and property damage. Large businesses may fall victim to various schemes.
2.	Idea / Solution description	A significant area for investigation is the detection and prevention of phishing websites. There are several phishing strategies that provide attackers with numerous and vital ways to access the information of individuals and companies. Locator for uniform resources URLs, which are occasionally referred to as "Web links," are crucial in phishing attacks. Uniform resource locator has the potential to reroute pages, i.e., through a hyperlink, to either a legitimate website or a phishing website. Every day, new methods for creating phishing sites appear. This in fact encouraged a number of researchers to focus their efforts on locating the phishing sites.
3.	Novelty / Uniqueness	We have thoroughly examined and determined several elements that may be utilized to recognize a phishing website. These elements can be classified as address bar-based features, domain-based features, HTML-based features, or JavaScript-based features. We can accurately detect a phishing site using these qualities.
4.	Social Impact / Customer Satisfaction	Each indexing database was subjected to a thorough, methodical search. The most recent information on web phishing detections was gathered. On the basis of the methodology, the papers were categorized. By doing a thorough scan on the top-secret documents, taxonomy was developed. The contributions described in this survey are comprehensive and include every recent advancement in this field.
5.	Business Model (Revenue Model)	Spreading fraudulent email is the first step in a phishing scan. When an online user clicks on a phishing URL or link, anti-phishing mechanisms begin to act, either by forwarding the phishing email to the spam folder or by displaying a warning. the stages of a phishing attack in this region.
6.	Scalability of the Solution	The domains of phishing sites show th characteristics that set them apart from othe websites and domains. (Google, www.google.com and any other random phishing website www.google.com, are two examples.) Whe compared to other websites and domain names phishing Uniform Resource Locators and "domai names" frequently differ in length. This affects bot the training accuracy and testing accuracy of all th models. The variance in train and test accurac numbers demonstrates that the models are no being overfit to the extensive dataset.

Fig. 3.3 Proposed Solution

3.4. PROBLEM SOLUTION FIT

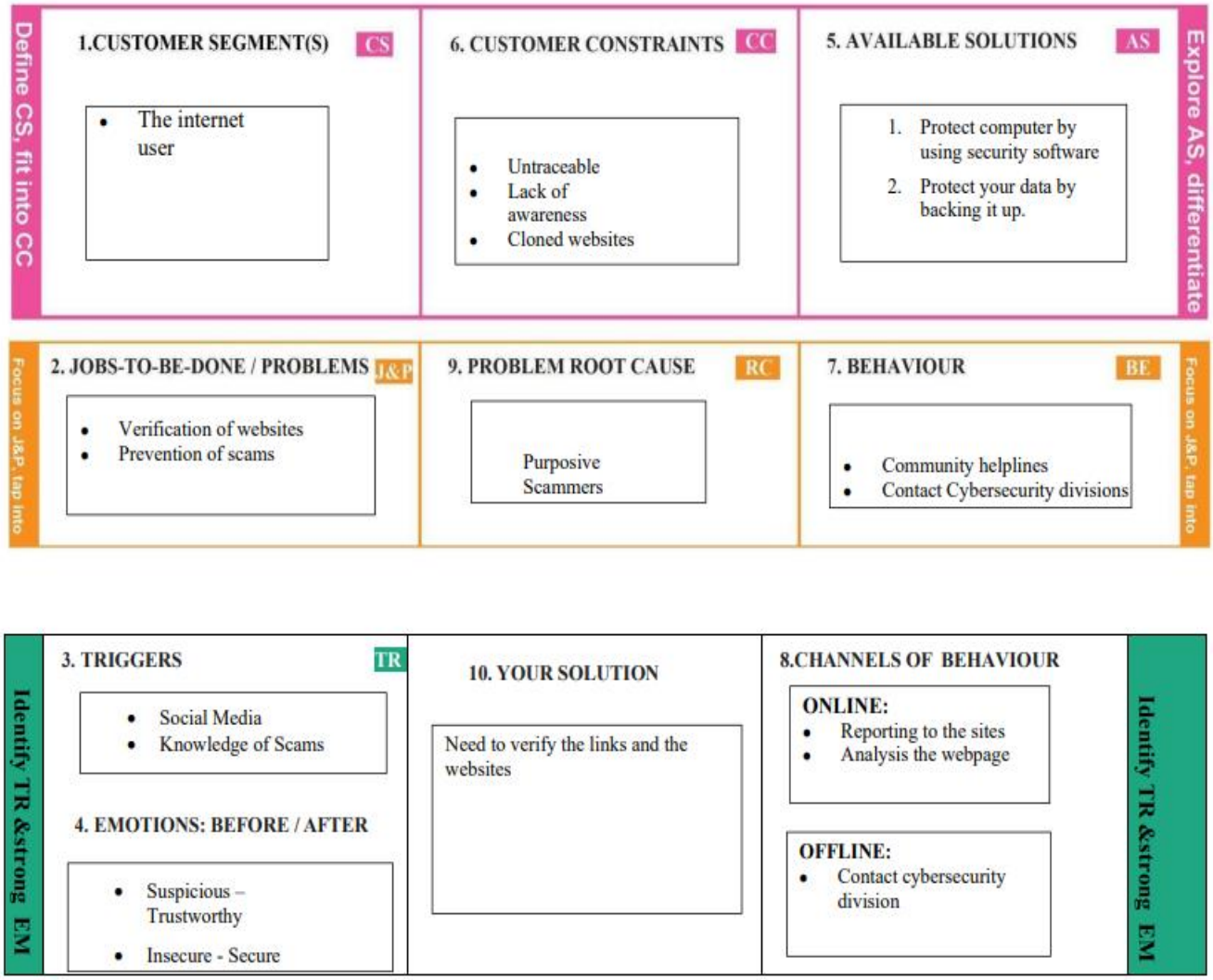


Fig. 3.4 Problem Solution Fit

CHAPTER 4

REQUIREMENT ANALYSIS

4.1. FUNCTIONAL REQUIREMENT

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	User Authentication	Confirmation of Google Firebase
FR-4	User Security	Strong Passwords, 2FA and FIDO2.0
FR-5	User Performance	Usage of Legitimate websites, Optimize Network Traffic

Fig. 4.1.Functional Requirements

4.2. NON-FUNCTIONAL REQUIREMENT

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It is incredibly user-friendly, and through our alert message, even those with limited knowledge may recognize when they are accessing a bogus website.
NFR-2	**Security**	One may simply trust our detection website because it cannot be hacked, and they will be protected from financial and information loss as a result.
NFR-3	**Reliability**	It consistently performs well because it aggressively identifies phony websites, safeguards user data, and prevents financial loss.
NFR-4	**Performance**	Due to its ease of use, high level of security, and scalability, web phishing detection performs well and is quite effective.
NFR-5	**Availability**	This user-friendly detection website is accessible on any device, including desktop, laptop, and mobile.
NFR-6	**Scalability**	The fundamental concepts for scalable phishing detection and isolation are to shift end-user protection to the network provider and to apply the innovative "bad neighborhood" concept in order to identify and isolate both phishing emails.

Fig. 4.2.Non-Functional Requirements

CHAPTER 5

PROJECT DESIGN

5.1. DATAFLOW DIAGRAM

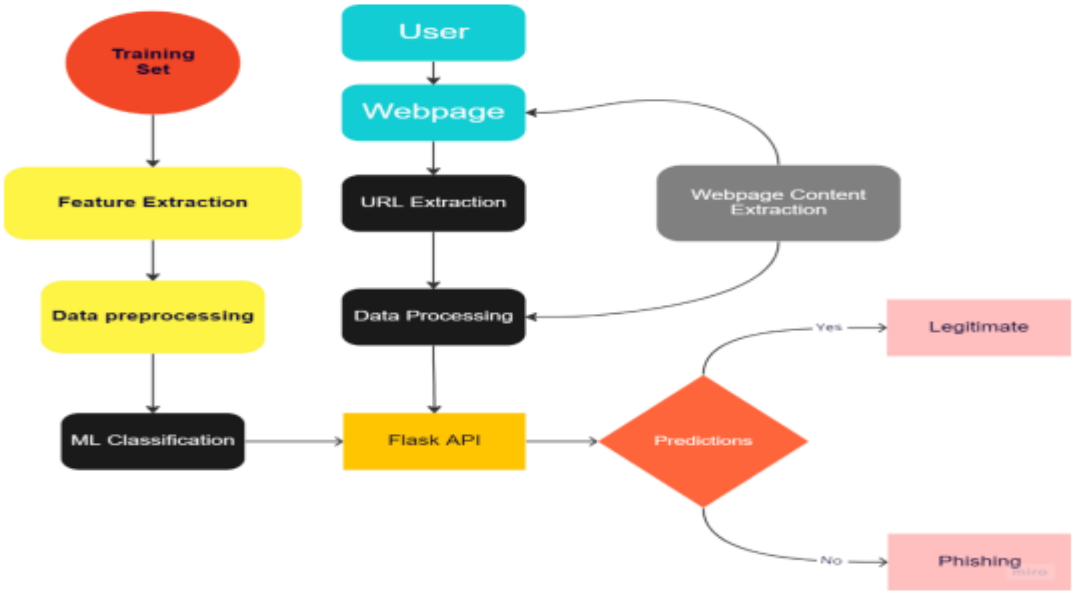


Fig. 5.1. Dataflow Diagram

5.2. TECHNICAL ARCHITECTURE

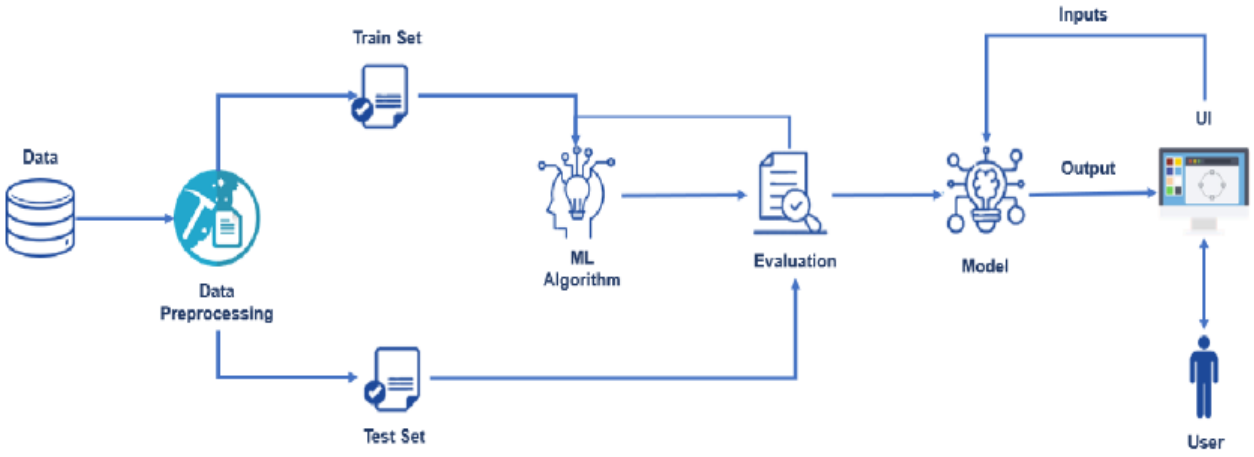


Fig. 5.2. Technical Architecture

5.3. SOLUTION ARCHITECTURE

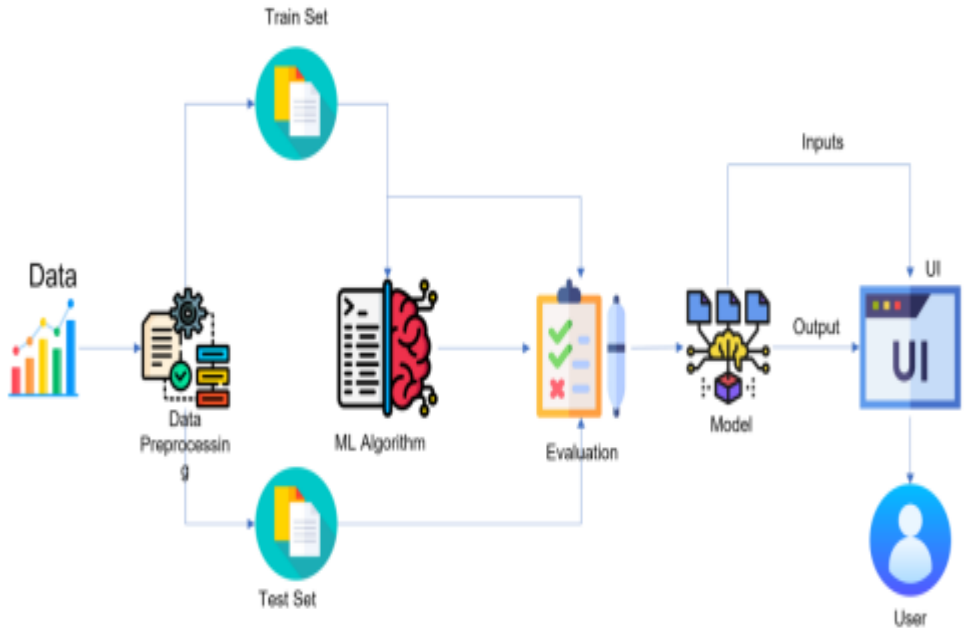


Fig. 5.3. Solution Architecture

5.4. USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard					
Customer (Web user)	User input	USN-1	As a user I can input the particular URL in the required field and waiting for validation	I can go access the website without any problem	High	Sprint-1
Customer Care Executive	Feature extraction	USN-1	After I compare in case if none found on comparison then we can extract feature using heuristic and visual similarity approach	As a user I can have comparison between websites for security.	High	Sprint-1
Administrator	Prediction	USN-1	Here the model will predict the URL website using Machine Learning algorithms such as Logistic Regression, KNN	In this I can have correct prediction on the particular algorithms	High	Sprint-1
	Classifier	USN-2	Here I will send all the model output to classifier in order to produce final result.	In this I will find the correct classifier for producing the result	Medium	Sprint-2

Fig. 5.4. User stories

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1. SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	User Input	USN-1	User inputs an URL in the required field to check its validation	1	Medium	VasanthKumar D
Sprint-1	Website Comparison	USN-2	Model compares the websites using Blacklist and Whitelist approach	1	High	Dharun S
Sprint-2	Feature Extraction	USN-3	After comparison, if none found on comparison then it extracts feature using heuristic and visual similarity	2	High	Abinash B
Sprint-2	Prediction	USN-4	Model predicts the URL using Machine learning algorithms such as logistic Regression, KNN.	1	Medium	Kishore G
Sprint-3	Classifier	USN-5	Model then displays whether the website is legal site or a phishing site.	1	Medium	Vasanthkumar D
Sprint-4	Announcement	USN-6	Model then displays whether the website is legal site or a phishing site	1	High	Abinash B
Sprint-4	Events	USN-7	This model needs the capability of retrieving and displaying accurate result for a website.	1	High	Dharun S

Fig. 6.1. Sprint planning & Estimation

6.2. SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
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	19 Nov 2022

Fig. 6.2. Sprint Delivery Schedule

CHAPTER 7

CODING & SOLUTION

7.1. FEATURE 1

7.1.1 DATA ANALYSIS

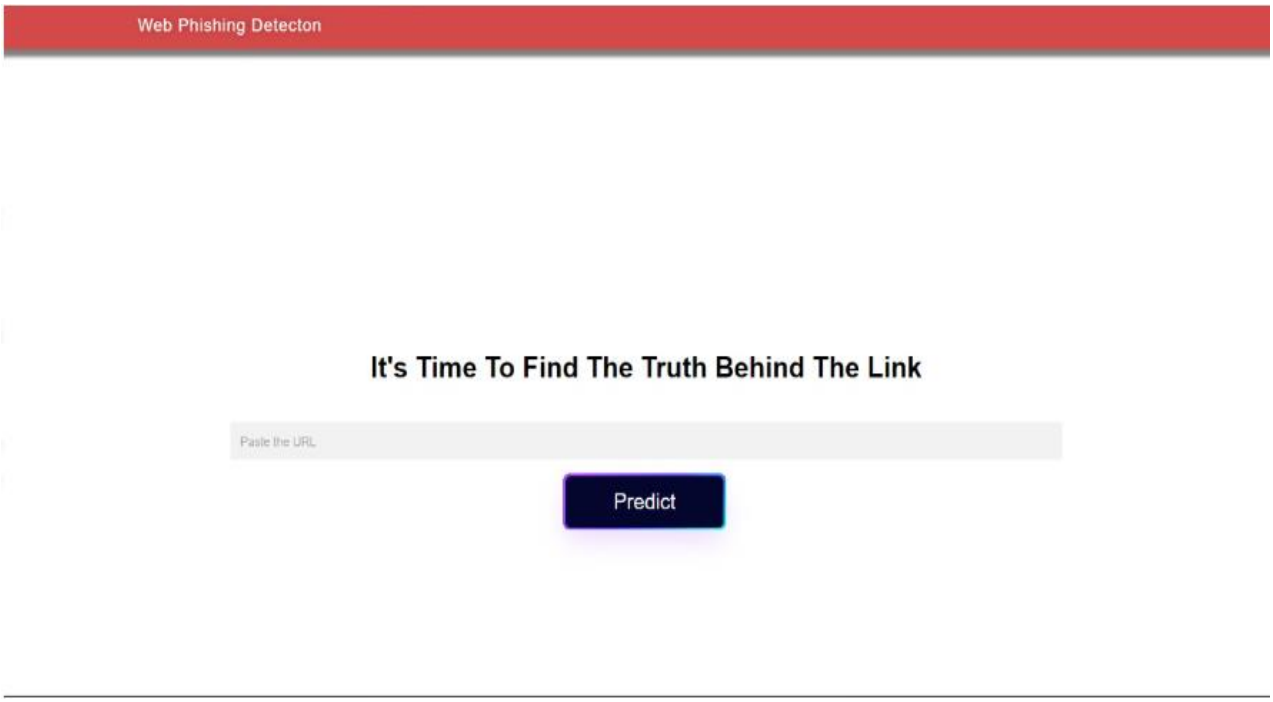
A method called data preprocessing is used to turn a raw data set into a clean data set. In other words, when data are gathered from various sources, they are gathered in raw form, which makes analysis impossible. It starts with the protocol used to visit the page is where it all starts. The server that is hosting the web page is identified by the fully qualified domain name. It consists of a top-level domain suffix and a registered domain name (second-level domain) (TLD). Due to the requirement that it be registered with a domain name Registrar, the domain name portion is limited. A domain name plus a subdomain name make up a host name. The subdomain sections are completely in the control of the phisher, who is free to assign any value. The path and file components of the URL are also possible, and they may both be altered at anytime by the phisher. The subdomain name and route are entirely under the phisher's control.

7.1.2 OVERVIEW OF DATA ANALYSIS

HOMEPAGE

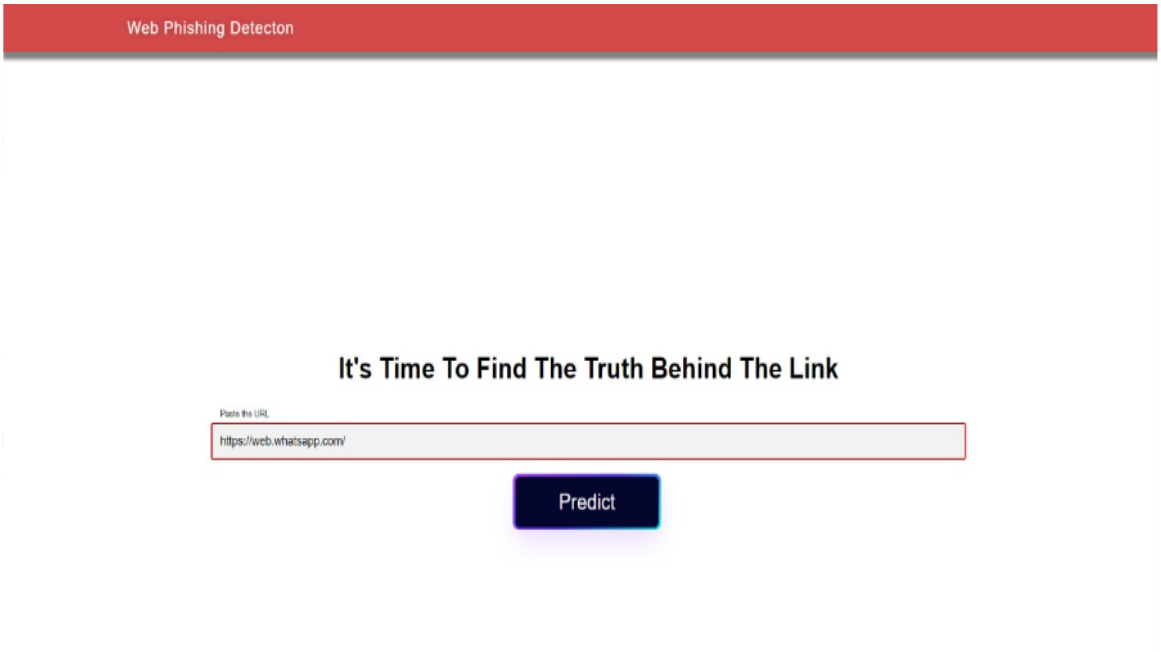
Fig 7.1.2.1 HOMEPAGE

URL PROVIDING



7.1.2.2. URL PROVIDING

URL PROVIDED



7.1.2.3 URL PROVIDED

7.2. FEATURE 2

7.2.1 CODING

```
import numpy as np

from flask import Flask, request, jsonify, render_template

import pickle

#importing the inputScript file used to analyze the URL

import inputScript

import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.

API_KEY = "nIRKSVDmk9sXH4oW1LtPgRbeaJMA8x0qJLtH2WFTt24L"

token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":

 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})

mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

#load model

app = Flask(_name_)

model = pickle.load(open('Phishing_Website.pkl', 'rb'))


@app.route('/')

def predict1():

    return render_template('index.html')

#Redirects to the page to give the user iput URL.
```

```
@app.route('/predict')
def predict():
    return render_template('final.html')
#Fetches the URL given by the URL and passes to inputScript
@app.route('/y_predict',methods=['POST'])
def y_predict():
    '''
    For rendering results on HTML GUI
    '''
    url = request.form['URL']
    checkprediction = inputScript.main(url)
    scoring = {"input_data": [{"field":
[["UsingIP","LongURL","ShortURL","Symbol@","Redirecting//","PrefixSuffix-
","SubDomains","HTTPS","DomainRegLen","Favicon","NonStdPort","HTTPSDomainURL","
RequestURL","AnchorURL","LinksInScriptTags","ServerFormHandler","InfoEmail","Abnorma
lURL","WebsiteForwarding","StatusBarCust","DisableRightClick","UsingPopupWindow","Ifra
meRedirection","AgeofDomain","DNSRecording","WebsiteTraffic","PageRank","GoogleIndex"
,"LinksPointingToPage","StatsReport"
                                ]], "values": checkprediction}]}
    response_scoring = requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/bf163e78-832a-470d-894f-
f7b8fbe4ac0d/predictions?version=2022-11-18', json=scoring,
    headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response")
    predictions = response_scoring.json()
```

```
    pred = predictions['predictions'][0]['values'][0][0]
    output=pred
    if(pred==1):
        pred="Your are safe!!  This is a Legitimate Website."
    else:
        pred="You are on the wrong site. Be cautious!"
    return render_template('final.html', prediction_text='{}'.format(pred),url=url)


#Takes the input parameters fetched from the URL by inputScript and returns the predictions
@app.route('/predict_api',methods=['POST'])
def predict_api():
    '''
    For direct API calls trought request
    '''
    data = request.get_json(force=True)
    prediction = model.y_predict([np.array(list(data.values()))])
    output = prediction[0]
    return jsonify(output)
if _name_ == "_main_":
    app.run(debug=True)
if _name_ == '_main_':
    app.run(host='0.0.0.0', debug=True)
```

7.2.2 OUTPUT

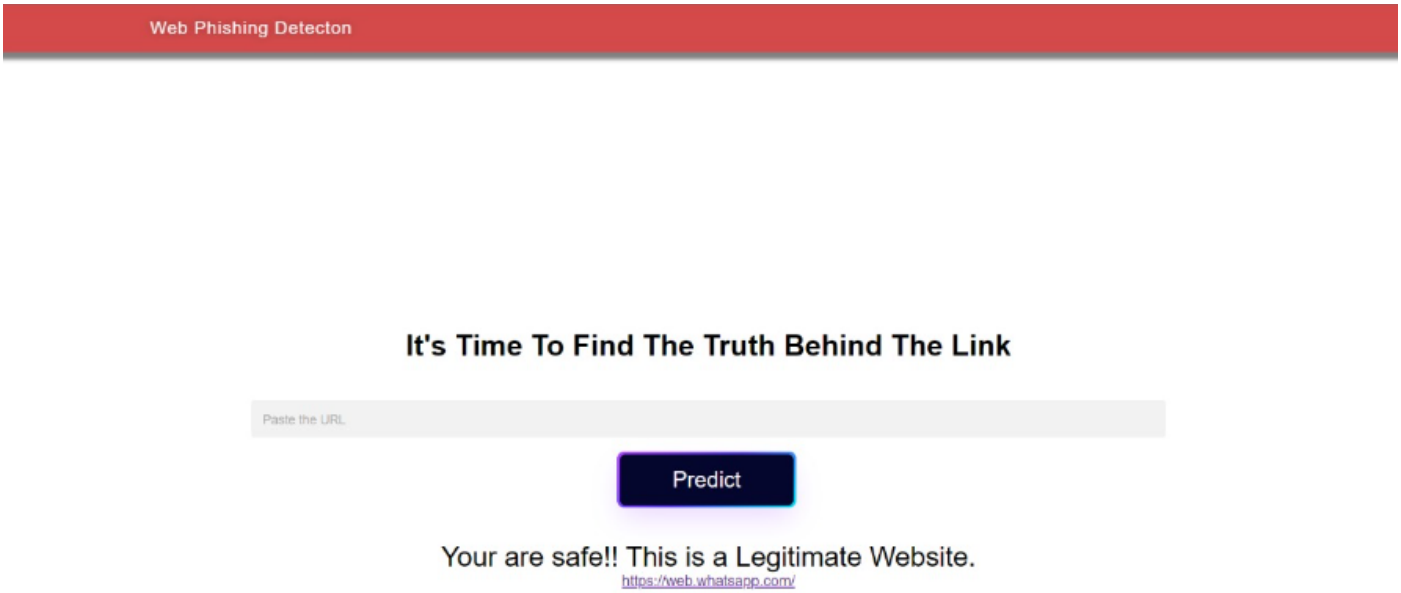


Fig 7.2.2.1 RESULT PAGE

CHAPTER 8

TESTING

8.1. TEST CASES

A test case is a series of operations carried out on a system to examine software compliance and proper operation. A test case's objective is to ascertain if various system features operate as anticipated and to verify that the system complies with all applicable standards, directives, and user requirements. A excellent technique to identify issues or flaws with the system is to create a test case. Test cases, which serve as detailed instructions for each system test, are typically created by members of the testing team or the quality assurance (QA) team. Once a system feature or group of features has been completed by the development team, testing may start. A test suite is a grouping or collection of test cases.

8.1.1 WHITE BOX TESTING

With the aid of the method known as "white box testing," testers can check and confirm all aspects of a software system's internal workings, including its infrastructure, code, and connections to external systems.

```
[main]  INFO    profile include tests: None
[main]  INFO    profile exclude tests: None
[main]  INFO    cli include tests: None
[main]  INFO    cli exclude tests: None
[main]  INFO    running on Python 3.8.6
[node_visitor]  WARNING Unable to find qualified name for module: app.py
Run started:2022-11-19 07:32:51.232240

Test results:
        No issues identified.

Code scanned:
        Total lines of code: 49
        Total lines skipped (#nosec): 0

Run metrics:
        Total issues (by severity):
                Undefined: 0
                Low: 0
                Medium: 0
                High: 0
        Total issues (by confidence):
                Undefined: 0
                Low: 0
                Medium: 0
                High: 0
Files skipped (0):
```

Fig 8.1.1.1 TESTING PYTHON USING BANDIT

8.1.2 BLACK BOX TESTING

A system is only examined from the outside during black box testing. The fundamental workings of the system that cause it to respond to test inputs are unknown to the tester and the operator. A system is referred to as a "black box" if the only way to comprehend its behaviour is by looking at its inputs and outputs.

8.2. USER ACCEPTANCE TESTING

User acceptance testing, or UAT, is a type of testing that the end user or client performs prior to deploying the software programme to a production environment. UAT is conducted as the final phase of testing following the conclusion of functional, integration, and system testing.

8.2.1 PURPOSE OF USER ACCEPTANCE TESTING

The main objective of UAT is to validate the whole business process. Minor typos, misspellings, or system testing are not its primary focus. Utilizing production-like data produced in a different testing environment, user acceptance testing is carried out. There will be two or more end users, and it will resemble black box testing.

8.2.2 NEEDS OF USER ACCEPTANCE TESTING

User Acceptance Testing is required after Unit, Integration, and System testing of the software because it is possible that developers built the software based on requirements documents in accordance with their own understanding and that additional changes that were necessary during development were not effectively communicated to them. Due to this, it is essential to verify whether the client or end-user would accept the final result.

CHAPTER 9

RESULTS

9.1. PERFORMANCE METRICS

Performance metrics are described as numbers and information that depict how well a business operates, what it is capable of, and how well it performs overall. The many various performance measures that may be employed include sales, profit, ROI, customer satisfaction, customer reviews, personal reviews, general quality, and reputation in the market. When compared between several industries, performance measures might differ dramatically. Performance indicators are essential to a business's success. Since these metrics aid in directing and evaluating an organization's success, businesses must choose their key performance indicators and concentrate on these areas. Important success elements can only be useful if they are recognised and monitored. In order for business measures to provide accurate findings and for the relevant questions to be posed, attentive management is also required.

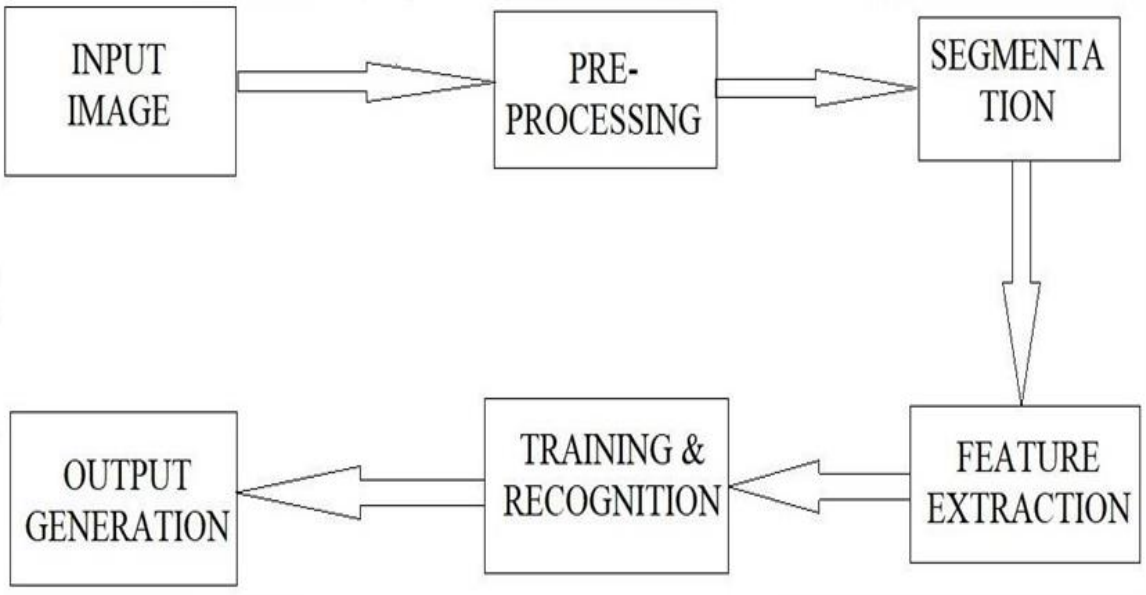


Fig 9.1.1 PERFORMANCE METRICS

CHAPTER 10

ADVANTAGES &DISADVANTAGES

ADVANTAGES

❖ The blacklist's three main elements are timing, quality, and quantity (the number of URLs in the list). PwdHash helps prevent password theft.

❖ Using SpoofGuard, unauthorised IP and MAC addresses are prevented. For Bayesian content filtering, a user-specific training set is available.

❖ It steers clear of false positives.

❖ URL verification is one of the numerous advantages of three-factor authentication.

❖ According to a credible source, URL verification stops 79% of phishing attacks.

❖ Using Open ID reduces the amount of time it takes to detect phishing attacks.

❖ Increasing public awareness of security-related issues of all types, including phishing, is always a good idea.

DISADVANTAGES

❖ By employing the "Bayesian poisoning" technique, con artists thwart Bayesian content screening.

❖ On occasion, scammers will alter the language to get around the database of the filter. For example, they might substitute "Viagra" with "Viaagra."

❖ Sometimes, this kind of service results in false positives. The blacklist's updating process can occasionally be slow, so a new phishing website could be dangerous since it has not yet been added.

❖ Unreliability Site target accepts the password that was collected.

❖ Employees may view this type of repeated training as a chore and frequently neglect it.

CHAPTER 11

CONCLUSION

The purpose of this work was to investigate the feasibility of detecting phishing attacks by separating, using CNNs, the URL and pictures of phishing websites from the URL and photos of authentic websites. Newly produced phishing webpages may be identified using the method we presented, which is based just based on the URL and screenshot of dubious websites. 99.67% classification accuracy is displayed by the suggested model.A conclusion that may be drawn from the data is that integrating URLs characteristics and visual Convolution neural networks are a superior method for detecting similarity since they are more effective than several methods for accomplishing automated feature extraction and categorization distinguishing between authentic and fraudulent websites. Furthermore, it is obvious that growing batch sizes sizes causes the model's accuracy to decrease.We recommend finding a way to automatically identify the shortest URLs and the tiniest screenshot sizes for webpages in order to improve the model in future work and help the suggested method perform as well as possible.September 2020, Volume 12, Number 5 of the International Journal of Computer Networks & Communications (IJCNC).

CHAPTER 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. Additionally, in order to enhance the system's speed, we intend to investigate numerous phishing strategies that make advantage of HTML and JavaScript capabilities, Network-based features, Content-based features, and Lexical and Network-based aspects of web pages. We specifically extract information from URLs and subject them to different classifiers.

CHAPTER 13

APPENDIX

13.1. SOURCE CODE

App.py

```
import numpy as np
from flask import Flask, request, jsonify, render_template
import pickle
#importing the inputScript file used to analyze the URL
import inputScript
#load model
app = Flask(__name__)
model = pickle.load(open('Phishing_Website.pkl', 'rb'))
@app.route('/')
def predict1():
    return render_template('index.html')
#Redirects to the page to give the user iput URL.
@app.route('/predict')
def predict():
    return render_template('final.html')
#Fetches the URL given by the URL and passes to inputScript
@app.route('/y_predict',methods=['POST'])
def y_predict():
    '''
    For rendering results on HTML GUI
    '''
    url = request.form['URL']
    checkprediction = inputScript.main(url)
    prediction = model.predict(checkprediction)
```

```
    print(prediction)
    output=prediction[0]
    if(output==1):
        pred="Your are safe!!  This is a Legitimate Website."

    else:
        pred="You are on the wrong site. Be cautious!"
    return render_template('final.html', prediction_text='{}'.format(pred),url=url)

#Takes the input parameters fetched from the URL by inputScript and returns the predictions
@app.route('/predict_api',methods=['POST'])
def predict_api():
    '''
    For direct API calls trought request
    '''
    data = request.get_json(force=True)
    prediction = model.y_predict([np.array(list(data.values()))])

    output = prediction[0]
    return jsonify(output)

if __name__ == "__main__":
    app.run(debug=True)

if __name__ == '__main__':
    app.run(host='0.0.0.0', debug=True)
```

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta content="width=device-width, initial-scale=1.0" name="viewport">
  <title>IBM Project</title>
  <meta content="" name="description">
  <meta content="" name="keywords">

  <!-- Google Fonts -->
  <link
href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,700i
|Jost:300,300i,400,400i,500,500i,600,600i,700,700i|Poppins:300,300i,400,400i,500,500i,600,60
0i,700,700i" rel="stylesheet">

  <!-- Vendor CSS Files -->
  <link href="assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <link href="assets/vendor/icofont/icofont.min.css" rel="stylesheet">
  <link href="assets/vendor/boxicons/css/boxicons.min.css" rel="stylesheet">
  <link href="assets/vendor/remixicon/remixicon.css" rel="stylesheet">
  <link href="assets/vendor/venobox/venobox.css" rel="stylesheet">
  <link href="assets/vendor/owl.carousel/assets/owl.carousel.min.css" rel="stylesheet">
  <link href="assets/vendor/aos/aos.css" rel="stylesheet">

<!--link href="{{ url_for('static', filename='/vendor/bootstrap/css/bootstrap.min.css') }}"
rel="stylesheet">
<link href="{{ url_for('static', filename='/vendor/icofont/icofont.min.css') }}" rel="stylesheet">
<link href="{{ url_for('static', filename='/vendor/boxicons/css/boxicons.min.css') }}"
rel="stylesheet">
<link href="{{ url_for('static', filename='/vendor/remixicon/remixicon.css') }}"
rel="stylesheet">
```

```
<link href="{{ url_for('static', filename='/vendor/venobox/venobox.css') }}" rel="stylesheet">
<link href="{{ url_for('static', filename='/vendor/owl.carousel/assets/owl.carousel.min.css') }}" rel="stylesheet">
<link href="{{ url_for('static', filename='/vendor/aos/aos.css') }}" rel="stylesheet"-->

  <!-- Template Main CSS File -->
  <link href="assets/css/style.css" rel="stylesheet">
  <!--link href="{{ url_for('static', filename='css/style.css') }}" rel="stylesheet"-->



</head>

<body>

  <!-- ======= Header ======= -->
  <header id="header" class="fixed-top ">
    <div class="container d-flex align-items-center">

      <h1 class="logo mr-auto"><a href="index.html">Web Phishing Detection</a></h1>
      <!-- Uncomment below if you prefer to use an image logo -->
      <!-- <a href="index.html" class="logo mr-auto"><img src="assets/img/logo.png" alt="" class="img-fluid"></a>-->

    </div>
  </header><!-- End Header -->

  <!-- ======= Hero Section ======= -->
  <section id="hero" class="d-flex align-items-center">

    <div class="container">
      <div class="row">
```

```
      <div class="col-lg-6 d-flex flex-column justify-content-center pt-4 pt-lg-0 order-2 order-lg-1" data-aos="fade-up" data-aos-delay="200">
        <h1>Let's Find The Phishing Sites          </h1>
        <h2>Don't Let Someone To Steal Your Data</h2>
        <div class="d-lg-flex">
          <a href="http://localhost:5000/predict" class="btn-get-started scrollto">Scan Now</a>
        </div>
      </div>
      <div class="col-lg-6 order-1 order-lg-2 hero-img" data-aos="zoom-in" data-aos-delay="200">
        <img src="https://media.tenor.com/SRy6HR5ibvsAAAAC/phishing-phisher.gif" class="img-fluid animated" alt="">
      </div>
    </div>
  </div>

 </section><!-- End Hero -->

 <main id="main">


   <!-- ======= About Us Section ======= -->
   <section id="about" class="about">
     <div class="container" data-aos="fade-up">

       <div class="section-title">
         <h2>About The Project</h2>
       </div>

       <div class="row content">
         <div class="col-lg-6">
```

There are a number of users who purchase products online and make payments through e-banking. There are e-banking websites that ask users to provide sensitive data such as username, password & credit card details, etc., often for malicious reasons. This type of e-banking website is known as a phishing website. Web service is one of the key communications software services for the Internet. Web phishing is one of many security threats to web services on the Internet.

In order to detect and predict e-banking phishing websites, we proposed an intelligent, flexible and effective system that is based on using classification algorithms. We implemented classification algorithms and techniques to extract the phishing datasets criteria to classify their legitimacy. The e-banking 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. Once a user makes a transaction online when he makes payment through an e-banking website our system will use a data mining algorithm to detect whether the e-banking website is a phishing website or not.

</section>

<footer id="footer">

```
  </footer>
  <a href="#" class="back-to-top"><i class="ri-arrow-up-line"></i></a>
  <div id="preloader"></div>
  <script src="assets/vendor/jquery/jquery.min.js"></script>
  <script src="assets/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
  <script src="assets/vendor/jquery.easing/jquery.easing.min.js"></script>
  <script src="assets/vendor/php-email-form/validate.js"></script>
  <script src="assets/vendor/waypoints/jquery.waypoints.min.js"></script>
  <script src="assets/vendor/isotope-layout/isotope.pkgd.min.js"></script>
  <script src="assets/vendor/venobox/venobox.min.js"></script>
  <script src="assets/vendor/owl.carousel/owl.carousel.min.js"></script>
  <script src="assets/vendor/aos/aos.js"></script>
  <script src="assets/js/main.js"></script>
 </body>
 </html>
```

Final.html

```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>IBM Project</title>
        <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet'
type='text/css'>
        <link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>
        <link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet'
type='text/css'>
        <link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300'
rel='stylesheet' type='text/css'>
        <link rel="stylesheet" href="{{ url_for('static', filename='css/final1.css') }}">
```

```
        <!--link rel="stylesheet" href="G:\Gayatri Files\Smartbridge\Nidhi\Phishing Website\static\css\style1.css"-->


    <style>
    .login{
    top: 20%;
    }
    </style>
    </head>
    <body>
    <div class="header">
    <div>Web Phishing Detecton</div>
    </div>
    <div class="main">
    <h1>It's Time To Find The Truth Behind The Link<h1>
    </div>
    <form action="{{ url_for('y_predict')}}"method="post">
            <label class="custom-field three">
            <input type="text" name="URL" placeholder=" " required/>
            <span class="placeholder">Paste the URL</span>
            <span class="border"></span>
           </label>
            <button id="button-64" role="button"><span class="text">Predict</span></button>
            </form>
            <br>
            <br>
            <div id='result',class='result' style='color:black;font size:30px;'>{{ prediction_text }}</div>
            <a href=" {{ url }} "> {{ url }} </a>
    </body>
    </html>
```

Style.css

```
@import url(https://fonts.googleapis.com/css?family=Open+Sans);

.btn { display: inline-block; *display: inline; *zoom: 1; padding: 4px 10px 4px; margin-bottom: 0; font-size: 13px; line-height: 18px; color: #333333; text-align: center;text-shadow: 0 1px 1px rgba(255, 255, 255, 0.75); vertical-align: middle; background-color: #f5f5f5; background-image: -moz-linear-gradient(top, #ffffff, #e6e6e6); background-image: -ms-linear-gradient(top, #ffffff, #e6e6e6); background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#ffffff), to(#e6e6e6)); background-image: -webkit-linear-gradient(top, #ffffff, #e6e6e6); background-image: -o-linear-gradient(top, #ffffff, #e6e6e6); background-image: linear-gradient(top, #ffffff, #e6e6e6); background-repeat: repeat-x; filter: progid:dximagetransform.microsoft.gradient(startColorstr=#ffffff, endColorstr=#e6e6e6, GradientType=0); border-color: #e6e6e6 #e6e6e6 #e6e6e6; border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25); border: 1px solid #e6e6e6; -webkit-border-radius: 4px; -moz-border-radius: 4px; border-radius: 4px; -webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05); -moz-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05); box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05); cursor: pointer; *margin-left: .3em; }

.btn:hover, .btn:active, .btn.active, .btn.disabled, .btn[disabled] { background-color: #e6e6e6; }

.btn-large { padding: 9px 14px; font-size: 15px; line-height: normal; -webkit-border-radius: 5px; -moz-border-radius: 5px; border-radius: 5px; }

.btn:hover { color: #333333; text-decoration: none; background-color: #e6e6e6; background-position: 0 -15px; -webkit-transition: background-position 0.1s linear; -moz-transition: background-position 0.1s linear; -ms-transition: background-position 0.1s linear; -o-transition: background-position 0.1s linear; transition: background-position 0.1s linear; }

.btn-primary, .btn-primary:hover { text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25); color: #ffffff; }

.btn-primary.active { color: rgba(255, 255, 255, 0.75); }

.btn-primary { background-color: #4a77d4; background-image: -moz-linear-gradient(top, #6eb6de, #4a77d4); background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4); background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#6eb6de), to(#4a77d4)); background-image: -webkit-linear-gradient(top, #6eb6de, #4a77d4); background-image: -o-linear-gradient(top, #6eb6de, #4a77d4); background-image: linear-gradient(top, #6eb6de, #4a77d4); background-repeat: repeat-x; filter: progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de, endColorstr=#4a77d4, GradientType=0);  border: 1px solid #3762bc; text-shadow: 1px 1px 1px rgba(0,0,0,0.4); box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.5); }

.btn-primary:hover, .btn-primary:active, .btn-primary.active, .btn-primary.disabled, .btn-primary[disabled] { filter: none; background-color: #4a77d4; }

.btn-block { width: 20%; display:block; margin-left:600px; }



* { -webkit-box-sizing:border-box; -moz-box-sizing:border-box; -ms-box-sizing:border-box; -o-box-sizing:border-box; box-sizing:border-box; }
```

```
html { width: 100%; height:100%; overflow:hidden; }

body {
                        width: 100vw;
                        height: 100vh;
                        display: flex;
                        flex-direction: column;
                        justify-content: center;
                        align-items: center;
                        font-family: sans-serif;
                        font-size: 16px;
  }
.header {
                                    top:0;
                                    margin:0px;
                                    left: 0px;
                                    right: 0px;
                                    position: fixed;
                                    background: #d44a4a;
                                    color: white;
                                    box-shadow: 0px 8px 4px grey;
                                    overflow: hidden;
                                    padding: 15px;
                                    font-size: 1.25vw;
                                    width: 100%;
                                    text-align: center;
                            }
.login {
                        position: absolute;
```

```
                top: 70%;
                left: 50%;
                margin: -25px 0 0 -150px;
                width:400px;
                height:400px;
}

  .header div { color: #fff; text-shadow: 0 0 10px rgba(0,0,0,0.3); letter-spacing:1px; text-
align:center; float:left; padding-left:150px;}

ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  padding-right:150px;
  overflow: hidden;
 }



li {
  float: right;
}

li a {
  display: block;
  color: white;
  text-align: center;
  padding: 0px 15px;
  text-decoration: none;
}
```

```
.main{
                        margin-top:200px;
                        color:black;

}

 *, *::before, *::after {
                        box-sizing: border-box;
 }

 .custom-field {
                        position: relative;
                        font-size: 14px;
                        border-top: 20px solid transparent;
                        margin-bottom: 5px;
                        display: inline-block;
                        --field-padding: 12px;
 }

 .custom-field input {
                        border: none;
                        -webkit-appearance: none;
                        -ms-appearance: none;
                        -moz-appearance: none;
                        appearance: none;
                        background: #f2f2f2;
                        padding: var(--field-padding);
                        border-radius: 3px;
                        width: 1000px;
                        outline: none;
```

```
                font-size: 14px;
}

.custom-field .placeholder {
                position: absolute;
                left: var(--field-padding);
                width: calc(100% - (var(--field-padding) * 2));
                overflow: hidden;
                white-space: nowrap;
                text-overflow: ellipsis;
                top: 22px;
                line-height: 100%;
                transform: translateY(-50%);
                color: #aaa;
                transition:
                  top 0.3s ease,
                  color 0.3s ease,
                  font-size 0.3s ease;
}

.custom-field input.dirty + .placeholder,
.custom-field input:focus + .placeholder,
.custom-field input:not(:placeholder-shown) + .placeholder {
                top: -10px;
                font-size: 10px;
                color: #222;
}

/* Link field */
.custom-field.three {
```

```
                --draw-duration: 0.1s;
                --draw-color: rgb(200, 0, 0);
                --draw-line-width: 2px;
                --draw-easing: linear;
}

.custom-field.three .border {
                width: 100%;
                height: 100%;
                position: absolute;
                left: 0;
                top: 0;
                transform: none;
                display: flex;
                align-items: center;
                padding-left: 12px;
                borde-radius: 3px;
}

.custom-field.three .border::after,
.custom-field.three .border::before {
                content: "";
                width: 0;
                height: 0;
                display: inline-block;
                position: absolute;
                border-radius: 3px;
}

.custom-field.three .border::before {
```

```
                left: 0;
                bottom: 0;
                border-right: 0px solid var(--draw-color);
                border-bottom: 0px solid var(--draw-color);
                transition:
                  border 0s linear calc(var(--draw-duration) * 4),
                  height var(--draw-duration) var(--draw-easing) calc(var(--draw-
duration) * 2),
                  width var(--draw-duration) var(--draw-easing) calc(var(--draw-
duration) * 3);
    }

    .custom-field.three .border::after {
                right: 0;
                top: 0;
                border-left: 0px solid var(--draw-color);
                border-top: 0px solid var(--draw-color);
                transition:
                  border 0s linear calc(var(--draw-duration) * 2),
                  height var(--draw-duration) var(--draw-easing),
                  width var(--draw-duration) var(--draw-easing) var(--draw-duration);
    }

    .custom-field.three input:focus ~ .border::before,
    .custom-field.three input:not(:placeholder-shown) ~ .border::before,
    .custom-field.three input.dirty ~ .border::before,
    .custom-field.three input:focus ~ .border::after,
    .custom-field.three input:not(:placeholder-shown) ~ .border::after,
    .custom-field.three input.dirty ~ .border::after {
                width: 100%;
                height: 100%;
```

```
                border-width: var(--draw-line-width);
    }

    .custom-field.three input:not(:placeholder-shown) ~ .border::before,
    .custom-field.three input.dirty ~ .border::before,
    .custom-field.three input:focus ~ .border::before {
                transition-delay: 0s, var(--draw-duration), 0s;
    }

    .custom-field.three input:not(:placeholder-shown) ~ .border::after,
    .custom-field.three input.dirty ~ .border::after,
    .custom-field.three input:focus ~ .border::after {
                transition-delay:
                  calc(var(--draw-duration) * 2),
                  calc(var(--draw-duration) * 3),
                  calc(var(--draw-duration) * 2);
    }
    #button-64 {
                margin-left: 40%;
                align-items: center;
                background-image: linear-gradient(144deg,#AF40FF, #5B42F3
50%,#00DDEB);
                border: 0;
                border-radius: 8px;
                box-shadow: rgba(151, 65, 252, 0.2) 0 15px 30px -5px;
                box-sizing: border-box;
                color: #FFFFFF;
                display: flex;
                font-family: Phantomsans, sans-serif;
                font-size: 20px;
                justify-content: center;
```

```
                        line-height: 1em;
                        max-width: 100%;
                        min-width: 140px;
                        padding: 3px;
                        margin-top: 10px;
                        text-decoration: none;
                        user-select: none;
                        -webkit-user-select: none;
                        touch-action: manipulation;
                        white-space: nowrap;
                        cursor: pointer;
}
#button-64:active,
#button-64:hover {
                        outline: 0;
}
#button-64 span {
                        background-color: rgb(5, 6, 45);
                        padding: 16px 24px;
                        border-radius: 6px;
                        width: 100%;
                        height: 100%;
                        transition: 300ms;
}
#button-64:hover span {
                        background: none;
}

@media (min-width: 768px) {
                        #button-64 {
```

```
                font-size: 24px;
                min-width: 196px;
            }
    }
```

13.2. GitHub Link & Project Demo Link

13.2.1 GitHub Link

Link : https://github.com/IBM-EPBL/IBM-Project-2753-1658482378

12.3. Project Demo Link

Project name: Web Phishing Detection

localhost:5000

Drive Link

https://drive.google.com/file/d/1L66sCJlkMBo1_shYaRU3qiew335FlNQB/view?usp=sharing

Youtube Link

https://youtu.be/2MWgpP-PGJM