

IBM NALAIYA THIRAN

PROJECT REPORT ON WEB PHISHING DETECTION

TEAM ID: PNT2022TMID39652

**C.ABDUL HAKEEM COLLEGE OF ENGINEERING
AND TECHNOLOGY**

Team Members

K.Hemapriya

H.Benazirfathima

S.Keerthana

J.Saranya

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1

7.2 Feature 2

7.3 Database Schema (if Applicable)

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

Source Code

GitHub & Project Demo Link

ABSTRACT

Phishing is the most commonly used social engineering and cyber attack. Through such attacks, the phisher targets naive online users by tricking them into revealing confidential information, with the purpose of using it fraudulently. In order to avoid getting phished, Users should have awareness of phishing websites. Have a blacklist of phishing websites which requires the knowledge of website being detected as phishing. Detect them in their early appearance, using machine learning and deep neural network algorithms. Of the above three, the machine learning based method is proven to be most effective than the other methods. A phishing website is a common social engineering method that mimics trustful uniform resource locators (URLs) and webpages. The objective of this project is to train machine learning models and deep neural nets on the dataset created to predict phishing websites. Both phishing and benign URLs of websites are gathered to form a dataset and from them required URL and website content-based features are extracted. The performance level of each model is measured and compared.

Keywords: Deep learning, Machine learning, Phishing website attack, Phishing website detection, Anti-phishing website, Legitimate website , Phishing website datasets, Phishing website features.

PRE-REQUISITES

TOOLS : JUPITER NOTEBOOK

OPERATING SYSTEM : WINDOWS 11

LANGUAGE : PYTHON

INSTALLING LIBRARIES

In this first step, we have to import the most common libraries used in python for machine learning such as

- Pandas
- Numpy
- Seaborn
- Matplotlib

IMPORTING DATA

In this project, we have used the url pre processed data.

CHAPTER 1

INTRODUCTION

Phishing imitates the characteristics and alternatives of emails and makes it appear similar due to the fact the original one. It seems nearly like that of the legitimate supply. The consumer thinks that this e-mail has come back from a real employer or a corporation. This makes the consumer to forcefully visit the phishing internet site thru the hyperlinks given inside the phishing email. These phishing web sites region unit created to mock the seams of an ingenious website. The phishers force person to inventory up the non-public info via giving baleful messages or validate account messages etc. so that they inventory up the preferred data which might be utilized by them to misuse it. They devise things such as the user isn't always left with the other choice but to go to their spoofed web site. Phishing is the most hazardous criminal physical activities in the cyber region. Since the maximum of the customers logs on to get admission to the services supplied with the aid of government and financial establishments, there has been a significant boom in phishing attacks for the beyond few years. Phishers commenced to earn cash and that they try this as a thriving business.

Phishing may be law-breaking, the explanation behind the phishers doing this crime is that it is terribly trustworthy to try to do this, it doesn't value something and it effective. The phishing will truly get entry to the e-mail identity of somebody it's terribly sincere to are looking for out the email identification currently every day and you will send an email to every person is freely offered throughout the globe. These attacker's vicinity terribly much less price and electricity to urge valuable know-how quick and truly. The phishing frauds effects malware infections, statistics loss, fraud, etc. information at some stage

in which those cyber criminals have an interest is that the crucial data of a user similar to the password, OTP, credit/ debit card numbers CVV, sensitive know- how associated with business, medical understanding, confidential information, etc commonly these criminals conjointly acquire data which may provide them directly get admission to do the social media account their emails.

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.

1.1 PROJECT OVERVIEW

- To develop a novel approach to detect malicious URL and alert users.
- To apply ML techniques in the proposed approach in order to analyze the real time URLs and produce effective results.
- To implement the concept of RNN, which is a familiar ML technique that has the capability to handle huge amount of data.

1.2 PURPOSE

- To develop an unsupervised deep learning method to generate insight from a URL.
- The study can be extended in order to generate an outcome for a larger network and protect the privacy of an individual.

CHAPTER 2

LITERATURE SURVEY:

TITLE: Phishing Detection from URLs Using Deep Learning

ApproachAUTHOR: Shweta Singh, M.P. Singh, Ramprakash Pandey

ABSTRACT:

Today, the Internet covers worldwide. All over the world, people prefer an E-commerce platform to buy or sell their products. Therefore, cybercrime has become the centre of attraction for cyber attackers in cyberspace. Phishing is one such technique where the unidentified structure of the Internet has been used by attackers/criminals that intend to deceive users with the use of the illusory website and emails for obtaining their credentials (like account numbers, passwords, and PINs). Consequently, the identification of a phishing or legitimate web page is a challenging issue due to its semantic structure. In this paper, a phishing detection system is implemented using deep learning techniques to prevent such attacks. The system works on URLs by applying a convolutional neural network (CNN) to detect the phishing webpage. In paper [19] the proposed model has achieved 97.98% accuracy whereas our proposed system achieved accuracy of 98.00% which is better than earlier model. This system doesn't require any feature engineering as the CNN extract features from the URLs automatically through its hidden layers. This is other advantage of the proposed system over earlier reported in [19] as the feature engineering is a very time-consuming task.

TITLE: WC-PAD: Web Crawling based Phishing Attack

DetectionAUTHOR: Nathezhtha, Sangeetha ,Vaidehi.

ABSTRACT:

Abstract- Phishing is a criminal offense which involves theft of user's sensitive data. The phishing websites target individuals, organizations, the cloud storage hosting sites and government websites. Currently, hardware-based approaches for Anti phishing are widely used but due to the cost and operational factors software-based approaches are preferred. The existing phishing detection approaches fails to provide solution to problem like zero-day phishing website attacks. To overcome these issues and precisely detect phishing occurrence a three-phase attack detection named as Web Crawler based Phishing Attack Detector (WC-PAD) has been proposed. It takes the web traffics, web content and Uniform Resource Locator (URL) as input features, based on these features classification of phishing and non-phishing websites are done. The experimental analysis of the proposed WC-PAD is done with datasets collected from real phishing cases. From the experimental results, it is found that the proposed WC-PAD gives 98.9% accuracy in both phishing and zero-day phishing attack detection

TITLE: A Deep Learning-Based Framework for Phishing Website

DetectionAUTHOR: LIZHEN TANG ,QUSAY H. MAHMOUD

ABSTRACT:

Phishing attackers spread phishing links through e-mail, text messages, and social media platforms. They use social engineering skills to trick users into visiting phishing websites and entering crucial personal information. In the end, the stolen personal information is used to defraud the trust of regular websites or financial institutions to obtain illegal benefits. With the development and applications of machine learning technology, many machine learning-based solutions for detecting phishing have been proposed. Some solutions are based on the features extracted by rules, and some of the features need to rely on third-party services, which will cause instability and time-consuming issues in the prediction service. In this paper, we propose a deep learning-based framework for detecting phishing websites. We have implemented the framework as a browser plug-in capable of determining whether there is a phishing risk in real-time when the user visits a web page and gives a warning message. The real-time prediction service combines multiple strategies to improve accuracy, reduce false alarm rates, and reduce calculation time, including whitelist filtering, blacklist interception, and machine learning (ML) prediction. In the ML prediction module, we compared multiple machine learning models using several datasets. From the experimental results, the RNN-GRU model obtained the highest accuracy of 99.18%, demonstrating the feasibility of the proposed solution.

TITLE: Phishing Detection in Websites using Parse Tree

ValidationAUTHOR: C. Emilin Shyni, Anesh D Sundar,
G.S.Edwin Ebby.

ABSTRACT:

Phishing is a technique of tricking people into giving sensitive information like usernames and passwords, credit card details, sensitive bank information, etc., by way of email spoofing, instant messaging, or using fake web sites whose look and feel gives the appearance of a legitimate website. In this work, a technique named parse tree validation is proposed to determine whether a webpage is legitimate or phishing. It is a novel approach to detect the phishing web sites by intercepting all the hyperlinks of a current page through Google API, and constructing a parse tree with the intercepted hyperlinks. This technique is implemented and tested with 1000 phishing pages and 1000 legitimate pages. The false negative rate achieved was 7.3% and the false positive rate achieved was 5.2%.

TITLE: Phishing Web Page Detection Methods: URL and HTML Features Detection

AUTHOR: Humam, Faris, Setiadi, Yazid.

ABSTRACT:

Phishing is a type of fraud on the Internet in the form of fake web pages that mimic the original web pages to trick users into sending sensitive information to phisher. The statistics presented by APWG and PhishTank show that the number of phishing websites from 2015 to 2020 tends to increase continuously. To overcome this problem, several studies have been carried out including detecting phishing web pages using various features of web pages with various methods. Unfortunately, the use of several methods is not really effective because the design and evaluation are only too focused on the achievement of detection accuracy in research, but evaluation does not represent application in the real world. Whereas a security detection device should require effectiveness, good performance, and deployable. In this study the authors evaluated several methods and proposed rules-based applications that can detect phishing more efficiently

TITLE: A Machine Learning Approach for URL Based Web Phishing Using Fuzzy Logic as Classifier

AUTHOR: Happy Chapla, Riddhi Kotak, Mittal Joiser.

ABSTRACT:

Phishing is the major problem of the internet era. In this era of internet the security of our data in web is gaining an increasing importance. Phishing is one of the most harmful ways to unknowingly access the credential information like username, password or account number from the users. Users are not aware of this type of attack and later they will also become a part of the phishing attacks. It may be the losses of financial found, personal information, reputation of brand name or trust of brand. So the detection of phishing site is necessary. In this paper we design a framework of phishing detection using URL.

CHAPTER 3

3.1 EXISTING PROBLEM

In this technological era, the Internet has made its way to become an inevitable part of our lives. It leads to many convenient experiences in our lives regarding communication, entertainment, education, shopping and so on. As we progress into online life, criminals view the Internet as an opportunity to transfer their physical crimes into a virtual environment. The Internet not only provides convenience in various aspects but also has its downsides, for example, the anonymity that the Internet provides to its users. Presently, many types of crimes have been conducted online. Hence, the main focus of our research is phishing. Phishing is a type of cybercrime where the targets are lured or tricked into giving up sensitive information, such as Social Security Number personal identifiable information and passwords. This obtainment of such information is done fraudulently. Given that phishing is a very broad topic, we have decided that this research should specifically focus on phishing websites.

Rao et al. [1] proposed a novel classification approach that use heuristic-based feature extraction approach. In this, they have classified extracted features into three categories such as URL Obfuscation features, Third-Party-based features, Hyperlink-based features. Moreover, proposed technique gives 99.55% accuracy. Drawback of this is that as this model uses third party features, classification of website dependent on speed of third-party services. Also this model is purely depends on the quality and quantity of the training set and Broken links feature extraction has a Volume 3.

Chunlin et al. [2] proposed approach that primarily focus on character frequency

features. In this they have combined statistical analysis of URL with machine learning technique to get result that is more accurate for classification of malicious URLs. Also they have compared six machine-learning algorithms to verify the effectiveness of proposed algorithm which gives 99.7% precision with false positive rate less than 0.4%. Sudhanshu et al. [3] used association data mining approach. They have proposed rule based classification technique for phishing website detection. They have concluded that association classification algorithm is better than any other algorithms because of their simple rule transformation. They achieved 92.67% accuracy by extracting 16 features but this is not up to mark so proposed algorithm can be enhanced for efficient detection rate.

M. Amaad et al.[4] presented a hybrid model for classification of phishing website. In this paper, proposed model carried out in two phase. In phase 1,they individually perform classification techniques, and select the best three models based on high accuracy and other performance criteria. While in phase 2, they further combined each individual model with best three model and makes hybrid model that gives better accuracy than individual model. They achieved 97.75% accuracy on testing dataset. There is limitation of this model that it requires more time to build hybrid model.

Hosseini et al.[5] developed an open-source framework known as “Fresh-Phish”. For phishing websites, machine-learning data can be created using this framework. In this, they have used reduced features set and using python for building query .They build a large labelled dataset and analyse several machine-learning classifiers against this dataset .Analysis of this gives very good accuracy using machine-learning classifiers. These analyses how long time it takes to train the model.

Gupta et al. [6] proposed a novel anti phishing approach that extracts features from client-side only. Proposed approach is fast and reliable as it is not dependent on third party but it extracts features only from URL and source code. In this paper, they have achieved 99.09% of overall detection accuracy for phishing website. This paper have concluded that this approach has limitation as it can detect webpage written in HTML .Non-HTML webpage cannot detect by this approach.

Bhagyashree et al.[7] proposed a feature based approach to classify URLs as phishing and nonphishing. Various features this approach uses are lexical features, WHOIS features, Page Rank and Alexa rank and Phish Tank-based features for disguising phishing and non-phishing website. In this paper, web-mining classification is used.

Mustafa et al.[8] developed safer framework for detecting phishing website. They have extracted URL features of website and using subset based selection technique to obtain better accuracy .In this paper, author evaluated CFS subset based and content based subset selection methods And Machine learning algorithms are used for classification purpose.

Priyanka et al.[9] proposed novel approach by combining two or more algorithms. In this paper ,author has implemented two algorithm Adaline and Backpropion along with SVM for getting good detection rate and classification purpose.

Pradeepthi et al.[10] In this paper ,Author studied different classification algorithm and concluded that tree-based classifier are best and gives better accuracy for phishing URL detection. Also Author uses various Volume 3, Issue 7, September-October-2018 | <http://ijsrcseit.com> Purvi Pujara et al. Int J S Res CSE & IT. 2018 September-October-2018; 3(7) : 395-399 398 features such as lexical features, URL based feature, network based

features and domain based feature.

Luong et al. [11] proposed new technique to detect phishing website. In proposed method, Author used six heuristics that are primary domain, sub domain, path domain, page rank, and alexa rank, alexa reputation whose weight and values are evaluated. This approach gives 97 % accuracy but still improvement can be done by enhancing more heuristics.

Ahmad et al.[12] proposed three new features to improve accuracy rate for phishing website detection. In this paper, Author used both type of features as commonly known and new features for classification of phishing and non-phishing site. At the end author has concluded this work can be enhanced by using this novel features with decision tree machine learning classifiers.

2.2 REFERENCES

- [1] Routhu Srinivasa Rao¹ , Alwyn Roshan Pais :Detection of phishing websites using an efficient feature-based machine learning framework :In Springer 2018. Volume 3, Issue 7, September-october-2018 | <http://ijsrcseit.com> Purvi Pujara et al. Int J S Res CSE & IT. 2018 September-October-2018; 3(7) : 395-399 399
- [2] Chunlin Liu, Bo Lang : Finding effective classifier for malicious URL detection : InACM,2018
- [3] Sudhanshu Gautam, Kritika Rani and Bansidhar Joshi : Detecting Phishing Websites Using Rule-Based Classification Algorithm: A Comparison : In Springer,2018.
- [4] M. Amaad Ul Haq Tahir, Sohail Asghar, Ayesha Zafar, Saira Gillani : A Hybrid Model to Detect Phishing-Sites using Supervised Learning Algorithms :In International Conference on Computational Science and Computational Intelligence IEEE ,2016.
- [5] Hossein Shirazi, Kyle Haefner, Indrakshi Ray: Fresh-Phish: A Framework for Auto-Detection of Phishing Websites: In (International Conference on Information Reuse and Integration (IRI)) IEEE,2017.
- [6] Ankit Kumar Jain, B. B. Gupta : Towards detection of phishing websites on client-side using machine learning based approach :In Springer Science+Business Media, LLC, part of Springer Nature 2017
- [7] Bhagyashree E. Sananse, Tanuja K. Sarode : Phishing URL Detection: A Machine Learning and Web Mining-based Approach : In International Journal of ComputerApplications,2015
- [8] Mustafa AYDIN, Nazife BAYKAL : Feature Extraction and Classification Phishing Websites Based on URL : IEEE,2015
- [9] Priyanka Singh, Yogendra P.S. Maravi, Sanjeev Sharma : Phishing Websites Detection through Supervised Learning Networks : In IEEE,2015
- [10] Pradeepthi. K V and Kannan. A: Performance Study of Classification Techniques for Phishing URL Detection: In 2014 Sixth International Conference on Advanced Computing(ICoAC) IEEE,2014
- [11] Luong Anh Tuan Nguyen[†], Ba Lam To[†] ,Huu Khuong Nguyen[†] and Minh Hoang Nguyen : Detecting Phishing Web sites: A Heuristic URL-Based Approach: In The 2013 International Conference on Advanced Technologies for Communications (ATC'13)
- [12] Ahmad Abunadi, Anazida Zainal ,Oluwatobi Akanb: Feature Extraction Process: A Phishing Detection Approach :In IEEE,2013.
- [13] Rami M. Mohammad, Fadi Thabtah, Lee McCluskey: An Assessment of Features

Related to Phishing Websites using an Automated Technique: In The 7th International Conference for Internet Technology and Secured Transactions, IEEE, 2012

Mohammad et al. [13] proposed model that automatically extracts important features for phishing website detection without requiring any human intervention. Author has concluded in this paper that the process of extracting feature by their tool is much faster and reliable than any manual extraction

2.3 PROBLEM STATEMENT DEFENETION

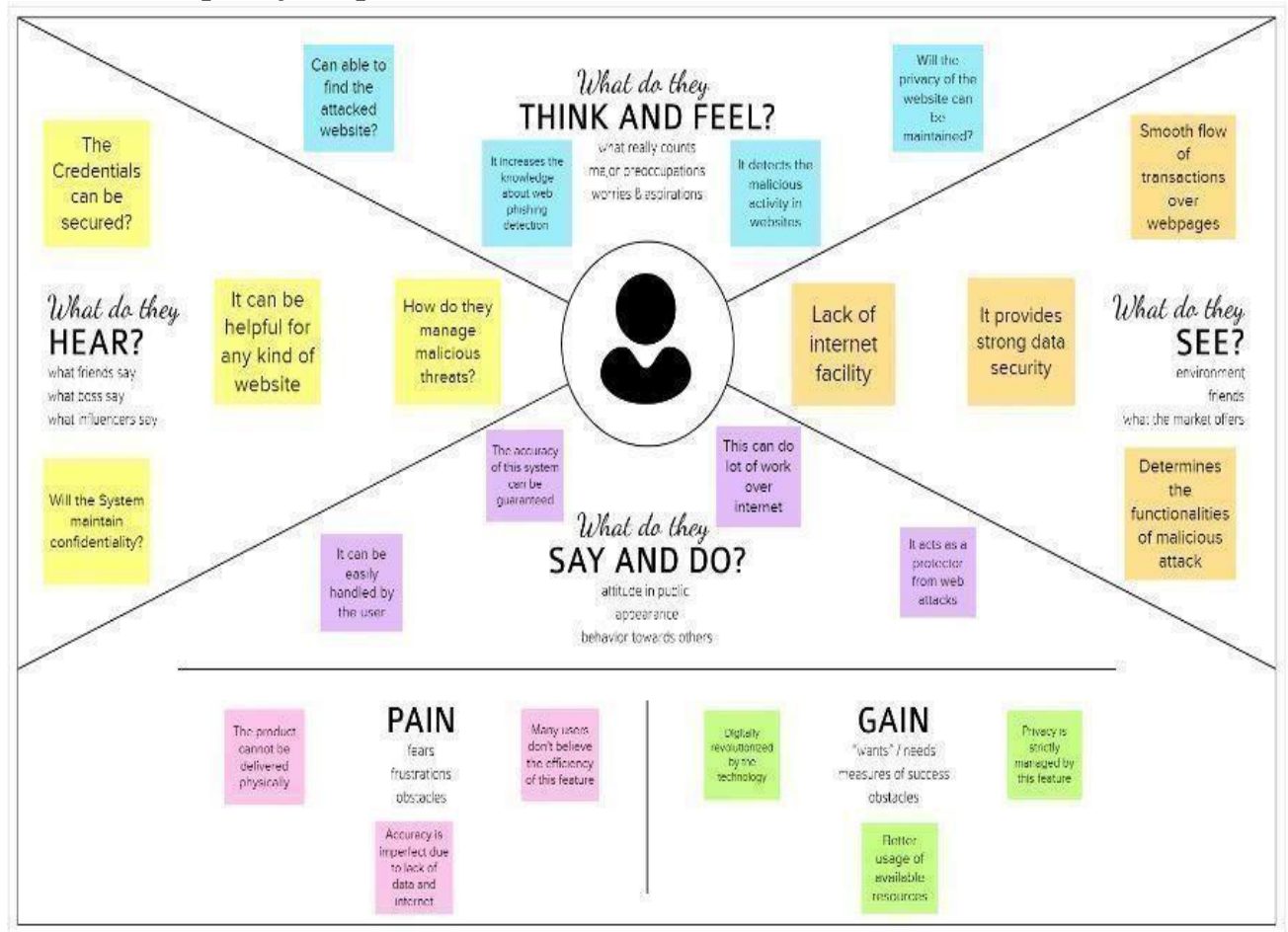
In order to detect and predict e-banking phishing websites, we proposed an intelligent, flexible and effective system that is based on using classification algorithms. We implemented classification algorithms and techniques to extract the phishing datasets criteria to classify their legitimacy. The e-banking 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.

Internet has dominated the world by dragging half of the world's population exponentially into the cyber world. With the booming of internet transactions, cybercrimes rapidly increased and with anonymity presented by the internet, Hackers attempt to trap the end-users through various forms such as phishing, SQL injection, malware, man-in-the-middle, domain name system tunnelling, ransomware, web trojan, and so on. Among all these attacks, phishing reports to be the most deceiving attack. Our main aim of this paper is classification of a phishing website with the aid of various machine learning techniques to achieve maximum accuracy and concise model.

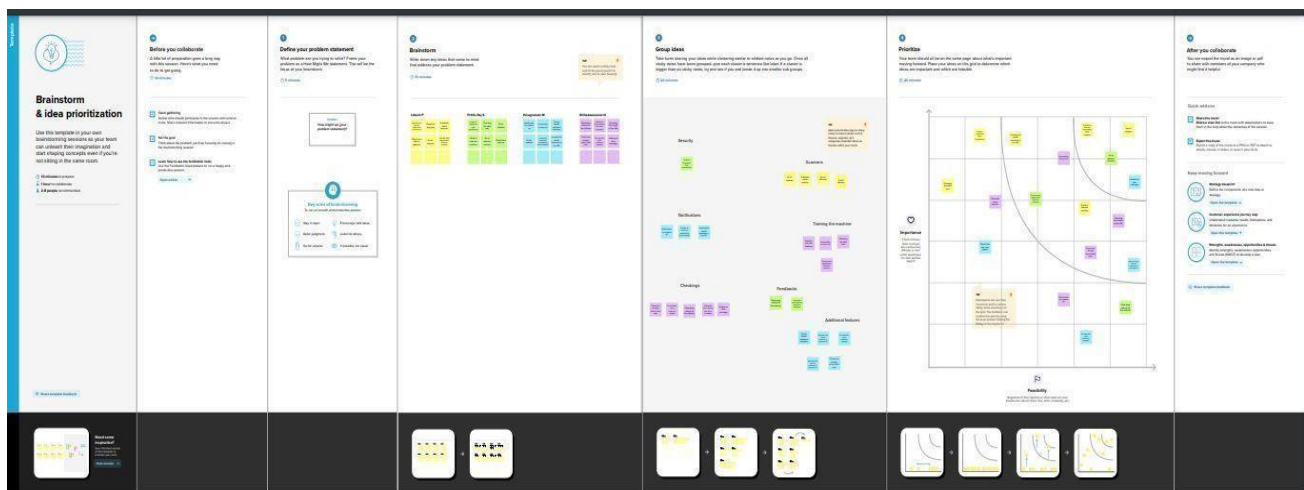
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming



3.3 Proposed Solution

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none">• Web phishing aims to steal private information, such as usernames, passwords, and credit card details, by way of impersonating a legitimate entity.• It will lead to information disclosure and property damage.• Large organizations may get trapped in different kinds of scams.
2.	Idea / Solution description	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.
3.	Novelty / Uniqueness	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.
4.	Social Impact / Customer Satisfaction	The feasibility of implementing this idea is moderate neither easy nor tough because the system needs to satisfy the basic requirements of the customer as well as it should act as a bridge towards achieving high accuracy on

		predicting and analysing the detected websites or files to protect our customer to the fullest.
5.	Business Model (Revenue Model)	People buy subscription annually, to protect their files both locally and at remote location with the help of our cloud integrated flask app for web phishing detection.
6.	Scalability of the Solution	By implementing this system, the people can efficiently and effectively to gain knowledge about the web phishing techniques and ways to eradicate them by detection. This system can also be integrated with the future technologies

3.4 Problem Solution fit:

Problem-Solution fit canvas 2.0 Purpose / Vision

1. CUSTOMER SEGMENT(S) <small>Who is your customer? i.e. working parents of 0-3 y.o. kids</small> CS Ecommerce Consumers	6. CUSTOMER CONSTRAINTS <small>What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices</small> CC ✓ Lack of awareness ✓ Untraceable scam websites ✓ Cloned websites	5. AVAILABLE SOLUTIONS <small>Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital, not tracking</small> AS ✓ Existing web phishing detection websites ✓ Word of Mouth ✓ News coverage ✓ Social Media
2. JOBS-TO-BE-DONE / PROBLEMS <small>Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one, explore different sides</small> JBP ✓ Authentication of websites ✓ Prevention of scams	9. PROBLEM ROOT CAUSE <small>What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations</small> RC ✓ Greedy Scammers ✓ Lack of awareness from customers	7. BEHAVIOUR <small>What does your customer do to address the problem and get the job done? i.e. directly related, find the right solar panel installer, calculate usage and benefits, indirectly associated, customers spend time on volunteering work (i.e. Greenpeace)</small> BE ✓ Contacting Cybersecurity ✓ Researching about website ✓ Web community helpline ✓ Reporting the site
3. TRIGGERS <small>What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a new efficient solution in the news</small> TR ✓ Reading about the E-Banking scams ✓ Social Media ✓ Past experiences	10. YOUR SOLUTION <small>If you are working on an existing business, write down your current solution first, fit it in the canvas, and check how much it fits really. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour</small> SL Verifies the genuineness of E-Banking websites/ Gateway	8. CHANNELS of BEHAVIOUR 8.1 ONLINE <small>What kind of actions do customers take online? Extract online channels from #7</small> CH ✓ Researching website ✓ Reporting the site 8.2 OFFLINE <small>What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development</small> ✓ Filing complaint with Bank ✓ Contacting Cybersecurity
4. EMOTIONS: BEFORE / AFTER <small>How do customers feel when they face a problem or a job and afterwards? i.e. fear, insecure > confident, in control > lost it in your communication strategy & design</small> EM ✓ Insecure > Secure ✓ Suspicious > Trustworthy		

Problem-Solution fit canvas is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 license
 Created by Daria Neprikladna / Amaltama.com

AMALTAMA

REQUIREMENT ANALYSIS

4.1 Functional Requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Verifying input	User inputs an URL (Uniform Resource Locator) innecessary field to check its validation.
FR-2	Website Evaluation	Model evaluates the website using Blacklist and Whitelist approach
FR-3	Extraction and Prediction	It retrieves features based on heuristics and visual similarities. The URL is predicted by the model using Machine Learning methods such as Logistic Regressionand KNN.
FR-4	Real Time monitoring	The use of Extension plugin should provide a warningpop-up when they visit a website that is phished. Extension plugin will have the capability to also detectlatest and new phishing websites
FR-5	Authentication	Authentication assures secure site, secure processes and enterprise information security.

4.2 Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

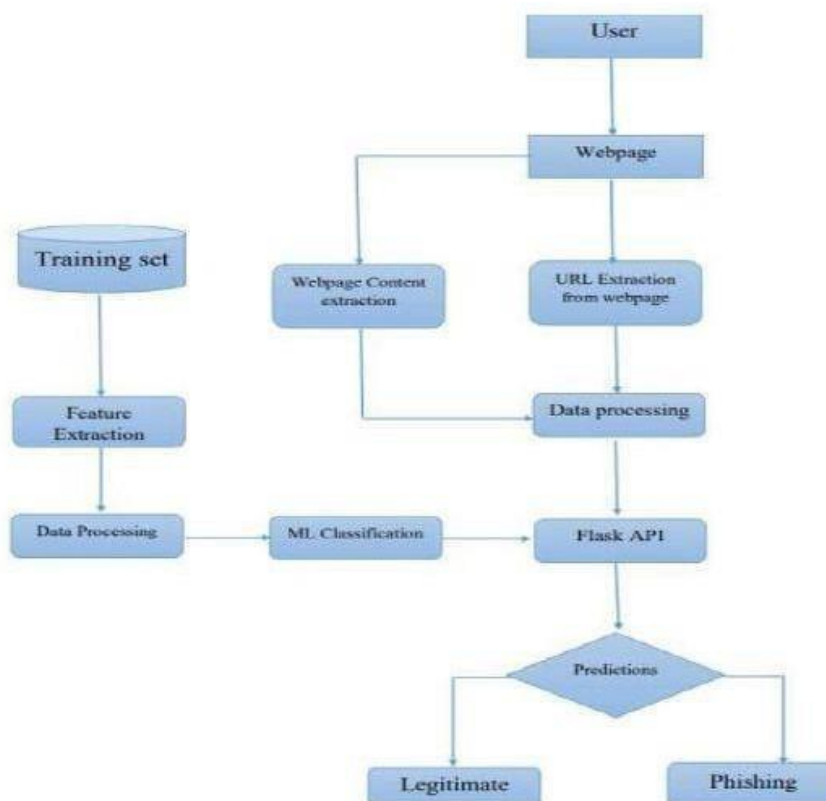
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Analysis of consumers' product usability in the design process with user experience as the core maycertainly help designers better grasp users' prospective demands in web phishing detection, behaviour, and experience.
NFR-2	Security	It guarantees that any data included within the system or its components will be safe from malwarethreats or unauthorised access. If you wish to prevent unauthorised access to the admin panel, describe the login flow and different user roles as system behaviour or user actions.
NFR-3	Reliability	It specifies the likelihood that the system or itscomponent will operate without failure for a specified amount of time under prescribed conditions.
NFR-4	Performance	It is concerned with a measurement of the system'sreaction time under various load circumstances.

NFR-5	Availability	It represents the likelihood that a user will be able to access the system at a certain moment in time. While it can be represented as an expected proportion of successful requests, it can also be defined as a percentage of time the system is operational within a certain time period.
NFR-6	Scalability	It has access to the highest workloads that will allow the system to satisfy the performance criteria. There are two techniques to enable the system to grow as workloads increase: Vertical and horizontal scaling.

PROJECT DESIGN

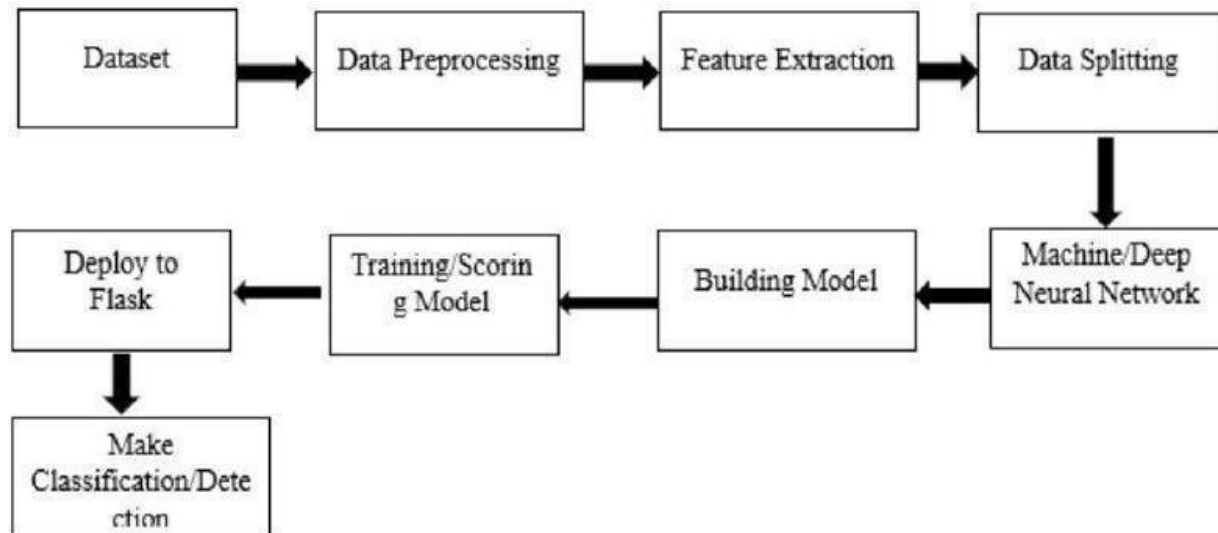
5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

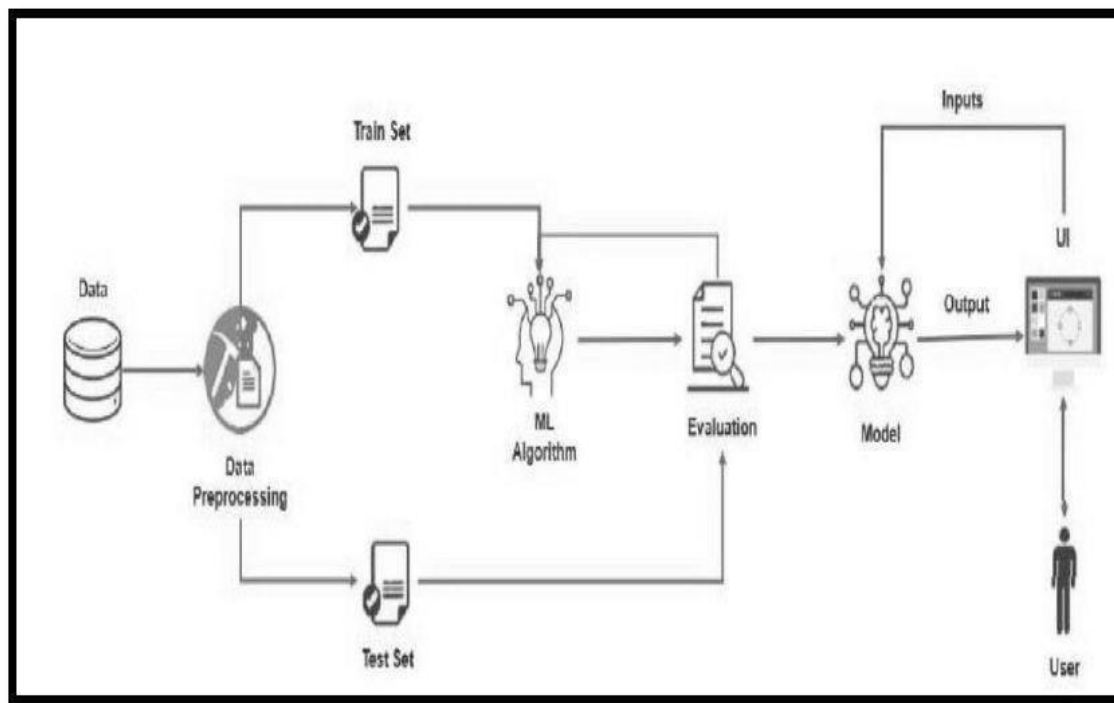


5.2 Solution and Technical Architecture

Solution Architecture



Technical Architecture: MODEL FOR WEB PHISHING DETECTION



5.3 USER STORIES

Use the below template to list all the user stories for the product.

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 websites 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.	I this i will find the correct classifier for producing the result	Medium	Sprint-2

PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Project Planning Phase

Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Date 01 November 2022
 Team ID PNT2022TMID39652
 Project Name Project – Web phishing Detection
 Maximum Marks 4 Marks

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

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	Saranya.j
Sprint-1	Website Comparison	USN-2	Model compares the websites using Blacklist and Whitelist approach.	1	High	Saranya.j
Sprint-2	Feature Extraction	USN-3	After comparison, if none found on comparison then it extract feature using heuristic and visual similarity.	2	High	Keerthana.S
Sprint-2	Prediction	USN-4	Model predicts the URL using Machine learning algorithms such as logistic Regression, KNN.	1	Medium	Keerthana.S
Sprint-3	Classifier	USN-5	Model sends all the output to the classifier and produces the final result.	1	Medium	Benazir fathima.H
Sprint-4	Announcement	USN-6	Model then displays whether the website is legal site or a phishing site.	1	High	Hemapriya.k
Sprint-4	Events	USN-7	This model needs the capability of retrieving and displaying accurate result for a website.	1	High	Hemapriya.K

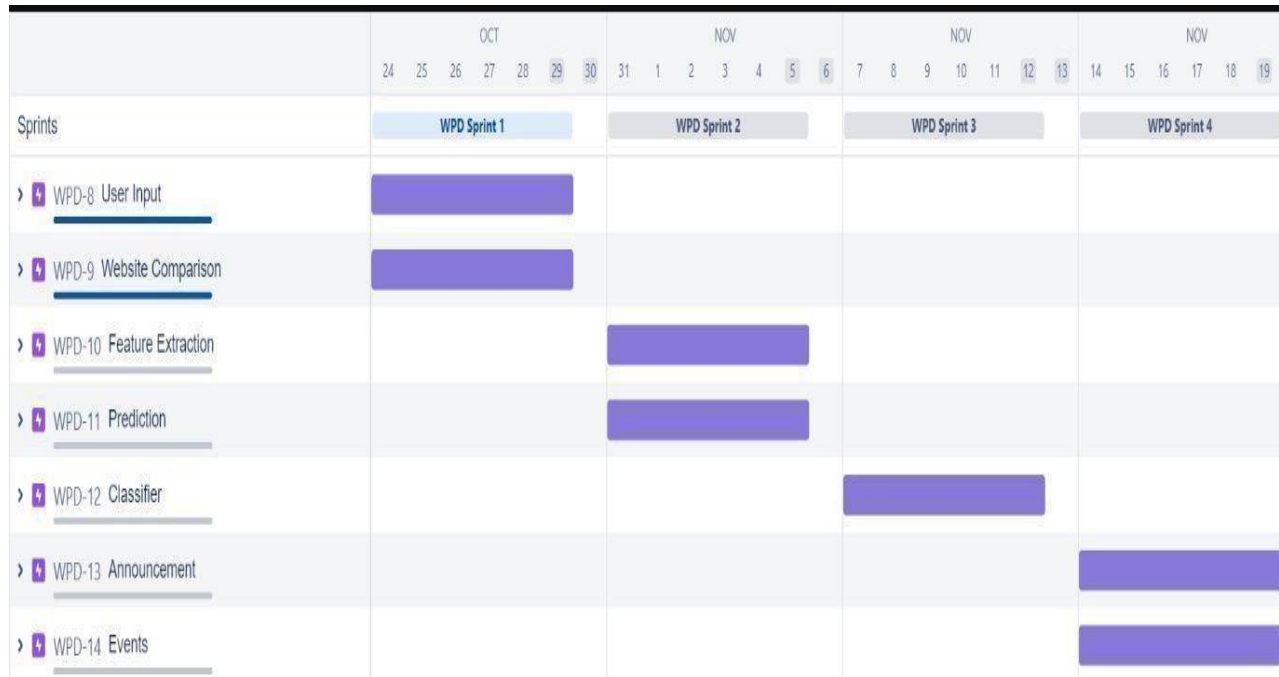
6.2 Sprint Delivery Schedule

PROJECT NAME:WEB PHISHING DETECTION

PROJECT ID :PNT2022TMID39652

SPRINT	TOTAL STORY POINTS	DURATION	SPRINT START DATE	SPRINT END DATE	STORY POINTS COMPLETED(AS ON PLANNED END DATE)	SPRINT RELEASE DATE(ACTUAL)
Sprint-1	20	5 Days	03 October 2022	07 October 2022	20	07 November 2022
Sprint-2	20	5 Days	08 October 2022	12 October 2022	20	07 November 2022
Sprint-3	20	5 Days	13 October 2022	17 October 2022	20	09 November 2022
Sprint-4	20	5 Days	18 October 2022	22 October 2022	20	10 November 2022

6.3 Reports from JIRA



CHAPTER-7

CODING & SOLUTION

7.1 Feature 1

```
# 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'))

#Redirects to the page to give the user input URL.
@app.route('/')
def predict():
    return render_template('final.html')

#Fetches the URL given by the user 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='{0}'.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 through 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(port='8000', debug=False)
```

CHAPTER 8

TESTING

8.1 Test Cases

				Date	15-Nov-22								
				Team ID	PNT 2022T MID03926								
				Project Name	Project-WebPhishing Detection								
				Milestone Marks	4marks								
Test case ID	Feature Type	Component	TestScenario	Pre-Requsite	Steps To Execute	TestData	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_OO 1	Functional	Home Page	Verify user is able to see the Landing Page when user can type the URL in the box		1.Enter URL and click go 2.Type the URL 3.Verify whether it is processing or not.	https://phishing-shield.herokuapp.com/	Should Display the Webpage	Workingas expected	Pass		N		S Balaji
LoginPage_TC_OO 2	UI	Home Page	Verify the UI elements is Responsive		1. Enter URL and click go 2. Type or copy paste the URL 3. Check whether the button is responsive or not 4. Reload and Test Simultaneously	https://phishing-shield.herokuapp.com/	Should Wait for Response and then get Acknowledge	Workingas expected	Pass		N		RABisheik
LoginPage_TC_OO 3	Functional	Home page	Verify whether the link is legitimate or not		1. Enter URL and click go 2. Type or copy paste the URL 3. Check the website is legitimate or not 4. Observe the results	https://phishing-shield.herokuapp.com/	User should observe whether the website is legitimate or not.	Workingas expected	Pass		N		TS Arwin
LoginPage_TC_OO 4	Functional	Home Page	Verify user is able to access the legitimate website or not		1. Enter URL and click go 2. Type or copy paste the URL 3. Check the website is legitimate or not 4. Continue if the website is legitimate or be cautious if its not legitimate.	https://phishing-shield.herokuapp.com/	Application should show that Safe Webpage or Unsafe.	Workingas expected	Pass		N		Balajee A V
LoginPage_TC_OO 5	Functional	Home Page	Testing the website with multiple URLs		1. Enter URL (https://phishing-shield.herokuapp.com/) and click go 2. Type or copy paste the URL to test 3. Check the website is legitimate or not 4. Continue if the website is secure or be cautious if its not secure	https://www.balajee.com/ https://www.google.com/ https://www.kinginfo.com/ https://www.delights.com/	User can able to identify the websites whether it is secure or not	Workingas expected	Pass		N		Balajee A V

8.2 User Acceptance Testing

UAT Execution & Report Submission

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [Web Phishing Detection] 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	10	2	4	20	36
Not Reproduced	0	0	1	0	1

Skipped	0	0	0	0	0
Won't Fix	0	0	2	1	3
Totals	23	9	12	25	60

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	10	0	0	10
Client Application	50	0	0	50
Security	5	0	0	4
Outsource Shipping	3	0	0	3
Exception Reporting	10	0	0	9
Final Report Output	10	0	0	10
Version Control	4	0	0	4

CHAPTER 9

RESULTS

9.1 Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Metrics	Classification Model: Gradient Boosting Classification Accuracy Score- 97.4%	
2.	Tune the Model	Hyperparameter Tuning - 97% Validation Method – KFOLD & Cross Validation Method	

1. METRICS:

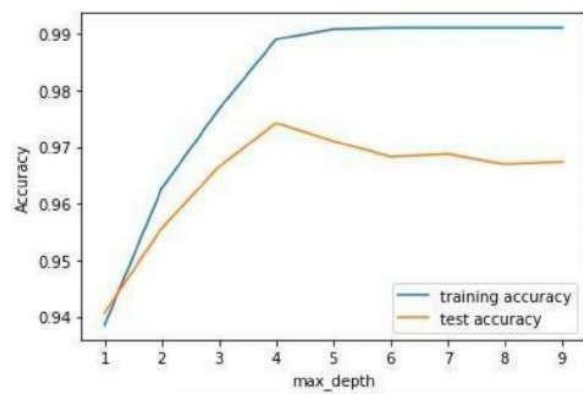
CLASSIFICATION REPORT:

In [52]: *#computing the classification report of the model*

```
print(metrics.classification_report(y_test, y_test_gbc))
```

	precision	recall	f1-score	support
-1	0.99	0.96	0.97	976
1	0.97	0.99	0.98	1235
accuracy			0.97	2211
macro avg	0.98	0.97	0.97	2211
weighted avg	0.97	0.97	0.97	2211

PERFORMANCE :



Out[83]:

	ML Model	Accuracy	f1_score	Recall	Precision
0	Gradient Boosting Classifier	0.974	0.977	0.994	0.986
1	CatBoost Classifier	0.972	0.975	0.994	0.989
2	Random Forest	0.969	0.972	0.992	0.991
3	Support Vector Machine	0.964	0.968	0.980	0.965
4	Decision Tree	0.958	0.962	0.991	0.993
5	K-Nearest Neighbors	0.956	0.961	0.991	0.989
6	Logistic Regression	0.934	0.941	0.943	0.927
7	Naive Bayes Classifier	0.605	0.454	0.292	0.997
8	XGBoost Classifier	0.548	0.548	0.993	0.984
9	Multi-layer Perceptron	0.543	0.543	0.989	0.983

2. TUNE THE MODEL – HYPERPARAMETER TUNING

```
In [58]: #HYPERPARAMETER TUNING  
grid.fit(X_train, y_train)
```

```
Out[58]: GridSearchCV  
GridSearchCV(cv=5,  
             estimator=GradientBoostingClassifier(learning_rate=0.7,  
                                                  max_depth=4),  
             param_grid={'max_features': array([1, 2, 3, 4, 5]),  
                        'n_estimators': array([ 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130,  
140, 150, 160, 170, 180, 190, 200])})  
└── estimator: GradientBoostingClassifier  
    GradientBoostingClassifier(learning_rate=0.7, max_depth=4)  
    └── GradientBoostingClassifier  
        GradientBoostingClassifier(learning_rate=0.7, max_depth=4)
```

```
In [59]: print("The best parameters are %s with a score of %.2f"  
             % (grid.best_params_, grid.best_score_))  
  
The best parameters are {'max_features': 5, 'n_estimators': 200} with a score of 0.97
```

VALIDATION METHODS: KFOLD & Cross Folding

Wilcoxon signed-rank test

```
In [78]: #KFOLD and Cross Validation Model

from scipy.stats import wilcoxon
from sklearn.datasets import load_iris
from sklearn.ensemble import GradientBoostingClassifier
from xgboost import XGBClassifier
from sklearn.model_selection import cross_val_score, KFold

# Load the dataset
X = load_iris().data
y = load_iris().target

# Prepare models and select your CV method
model1 = GradientBoostingClassifier(n_estimators=100)
model2 = XGBClassifier(n_estimators=100)
kf = KFold(n_splits=20, random_state=None)
# Extract results for each model on the same folds
results_model1 = cross_val_score(model1, X, y, cv=kf)
results_model2 = cross_val_score(model2, X, y, cv=kf)
stat, p = wilcoxon(results_model1, results_model2, zero_method='zsplit');
stat

Out[78]: 95.0
```

5x2CV combined F test

```
In [89]: from mlxtend.evaluate import combined_ftest_5x2cv
from sklearn.tree import DecisionTreeClassifier, ExtraTreeClassifier
from sklearn.ensemble import GradientBoostingClassifier
from mlxtend.data import iris_data

# Prepare data and clfs
X, y = iris_data()
clf1 = GradientBoostingClassifier()
clf2 = DecisionTreeClassifier()

# Calculate p-value
f, p = combined_ftest_5x2cv(estimator1=clf1,
                             estimator2=clf2,
                             X=X, y=y,
                             random_seed=1)

print('f-value:', f)
print('p-value:', p)

f-value: 1.727272727272733
p-value: 0.2840135734291782
```

CHAPTER -10

Advantages of web phishing detection

1. Improve on Inefficiencies of SEG and Phishing Awareness Training
2. It Takes a Load off the Security Team
3. It Offers a Solution, Not a Tool
4. Separate You from Your Competitors
5. This system can be used by many e-commerce websites in order to have good customer relationships.
6. If internet connection fails this system will work

Disadvantages of web phishing detection

1. All website related data will be stored in one place.
2. It is a very time-consuming process.

CHAPTER 11

CONCLUSION

It is outstanding that a decent enemy of phishing apparatus ought to anticipate the phishing assaults in a decent timescale. We accept that the accessibility of a decent enemy of phishing device at a decent time scale is additionally imperative to build the extent of anticipating phishing sites. This apparatus ought to be improved continually through consistent retraining. As a matter of fact, the accessibility of crisp and cutting-edge preparing dataset which may gained utilizing our very own device [30, 32] will help us to retrain our model consistently and handle any adjustments in the highlights, which are influential in deciding the site class. Albeit neural system demonstrates its capacity to tackle a wide assortment of classification issues, the procedure of finding the ideal structure is very difficult, and much of the time, this structure is controlled by experimentation. Our model takes care of this issue via computerizing the way toward organizing a neural system conspire; hence, on the off chance that we construct an enemy of phishing model and for any reasons we have to refresh it, at that point our model will encourage this procedure, that is, since our model will mechanize the organizing procedure and will request scarcely any client defined parameters.

CHAPTER-12

Future Scope

There is a scope for future development of this project. We will implement this using advanced deep learning method to improve the accuracy and precision. Enhancements can be done in an efficient manner. Thus, the project is flexible and can be enhanced at any time with more advanced features.

CHAPTER-13

Appendix:

1. Application Building
2. Collection of Dataset
3. Data Pre-processing
4. Integration of Flask App with IBM Cloud
5. Model Building
6. Performance Testing
7. Training the model on IBM
8. User Acceptance Testing
9. Ideation Phase
10. Preparation Phase
11. Project Planning
12. Performance Testing
13. User Acceptance Testing

Project Link: <https://github.com/IBM-EPBL/IBM-Project-18988-1659691966.git>