WEB PHISHING DETECTION

A PROJECT REPORT

Submitted by

SHAFIQUR RAHMAN(TL)

SATHISH KUMAR R

SHIYAM RAM N

VIJAYA PRASHANTH G S

of

COMPUTER SCIENCE AND ENGINEERING

ST. JOSEPH’S COLLEGE OF ENGINEERING

CHENNAI – 600119

CHAPTER 1

INTRODUCTION

1.1 Project Overview:

Be Safe is a website which is used to detect phishing websites to improve the customer's sense of safety whenever he/she attempts to provide any sensitive information to a site. Phishing is a form of fraud in which the attacker tries to learn sensitive information such as login credentials or account information by sending as a reputable entity or person in email or other communication channels. Also, by which people won't access them which will reduce the revenue of malicious site owners. This applicationcan be accessed online without paying instead, can be accessed via any browser of the customer'schoice to detect any site with high accuracy. This system uses machine learning algorithm which implements classification algorithms and techniques to extract the phishing datasets criteria to classify their legitimacy.

The design and implementation of a comprehensive web phishing detection system instils a cyber security culture which prevents the need for the deployment of targeted anti-phishing solutions in acorporate to meet industry's compliance obligations.

1.2 Purpose:

Web phishing is a threat in various aspects of security on the internet, which might involve scams andprivate information disclosure. Some of the common threats of web phishing are:

- Attempt to fraudulently solicit personal information from an individual or organization.
- Attempt to deliver malicious software by posing as a trustworthy organization or entity.
- Installing those malwares infects the data that cause a data breach or even nature's forcesthat takes down your company's data headquarters, disrupting access.

For this purpose, the objective of our project involves building an efficient and intelligent system to detect such websites by applying a machine-learning algorithm which implements classification algorithms and techniques to extract the phishing datasets criteria to classify their legitimacy and asa result of which whenever a user makes a transaction online and 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.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing problem:

There are phishing detection sites out in the web. But they charge users after a limit of usage. Most of them are built on a clean set of features. We have carefully analysed and identified several factors that could be used to detect a phishing site. These factors fall under the categories of address bar- based features, domain-based features, HTML & JavaScript based features. Using these features, we build an intelligent system which can identify a phishing site with high accuracy and efficiency. It is also an open-source website which will be easily accessible to all users.

2.2 References:

[1] Gunter Ollmann, "The Phishing Guide Understanding & Preventing Phishing Attacks", IBM Internet Security Systems, 2007.

[2] Mahmoud Khonji, "overview of phishing mitigation techniques", IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 15, NO. 4, FOURTH QUARTER 2013.

[3] Gaurav Varshney and Manoj Mishra, "web phishing detection schemes" in 'Security and communications networks ' pp 9:6266-6284 DOI: 10,1002/sec.1674 , 26 oct,2016

[4] Tewari presented a survey on "Fighting against Phishing attacks", Neural Computing and applications pp 3629–3654 (March17,2016). This paper simply provides a survey of current web anti-phishing solutions and fails to cover all the detection mechanisms available to mitigate the attack.

[5] Yoga Pristyanto and Akhmad Dahlan, "Hybrid Resampling for Imbalanced Class Handling on WebPhishing Classification Dataset", 2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), pp.401-406, 2019.

[6] Goel and Jain presented a survey about the mobile phishing attack and its life cycle. Volume 73 on "Computers & security" pp519-544,2018. However, this paper mainly focuses on the mobile phishing attack and its technical approaches only.

[7] Gaurav Varshney, Manoj Mishra and Pradeep K. Atrey, "A phish detector using lightweight searchfeatures", Computers & Security, 2016.

[8] Antonio Hernández Dominguez and Walter Baluja García, "Updated Analysis of Detection Methods for Phishing Attacks", Futuristic Trends in Network and Communication Technologies, vol.1395, pp.56, 2021.

2.3 Problem statement definition:

Web Phishing is a form of cyber fraud, which implies that fraudsters use various means to impersonate the URL address and page content of a real website or use vulnerabilities in the server program of a real website to insert dangerous HTML code in certain pages of the site.

It is a threat in various aspects of security on the internet, which might involve scams and private information disclosure. Some of the common threats of web phishing are:

- Obtaining personal information from an individual or organization.
- Impersonating as a trustworthy organization to deliver malicious websites.

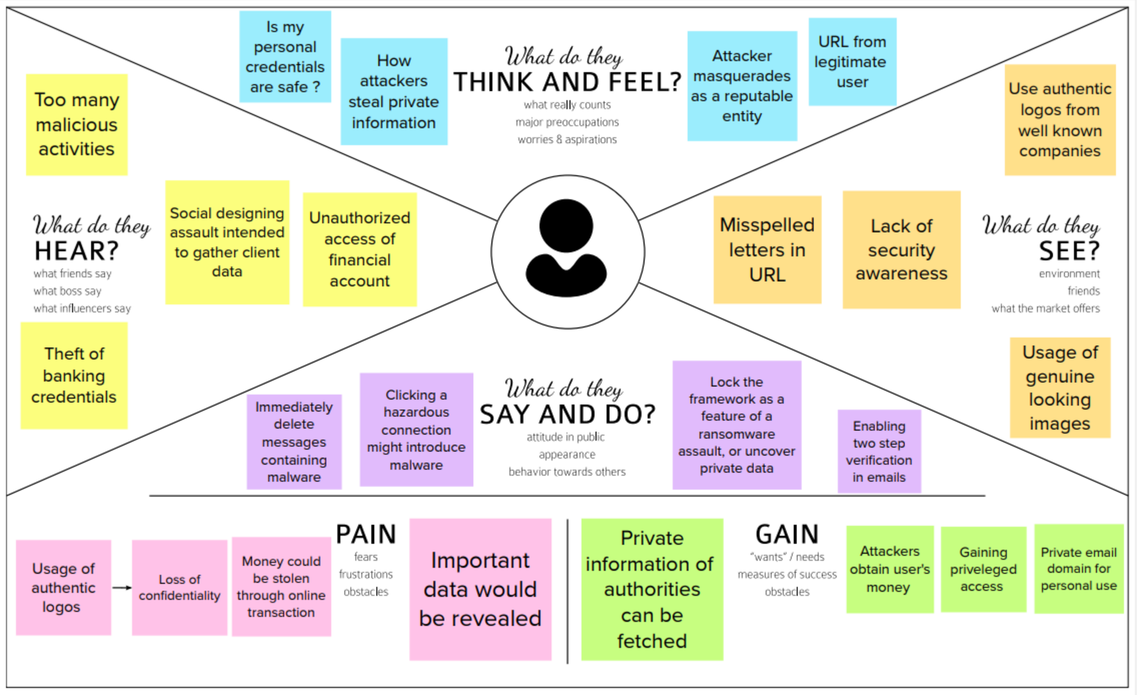
To avoid these threats, we build an efficient and intelligent system to detect such websites using machine-learning algorithms which implements classification algorithms and techniques to extract the phishing datasets criteria to classify their legitimacy.

This project can also be further extended by creating a browser extension or developing a GUI whichtakes the URL and analyses its nature to determine if it is a legitimate or a phishing website.

CHAPTER 3
IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:

An **empathy map** is a collaborative visualization used to articulate what we know about a particular type of user. It externalizes knowledge about users in order to create a shared understanding of user needs, and aid in decision making. Visualizing user attitudes and behaviors in an empathy map helps UX teams align on a deep understanding of end users. The mapping process also reveals any holes in existing user data. In user-centered design,empathy maps are best used from the very beginning of the design process.



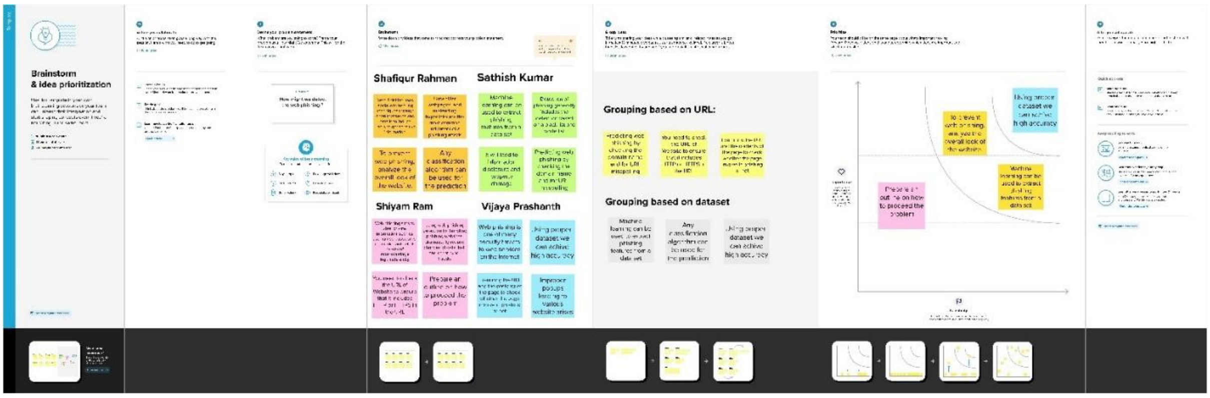
What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Is my personal credentials are safe ?
How attackers steal private information
Attacker masquerades as a reputable entity
URL from legitimate user
What do they HEAR?
what friends say
what boss say
what influencers say
Too many malicious activities
Social designing assault intended to gather client data
Unauthorized access of financial account
Theft of banking credentials
What do they SEE?
environment
friends
what the market offers
Use authentic logos from well known companies
Misspelled letters in URL
Lack of security awareness
Usage of genuine looking images
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
Immediately delete messages containing malware
Clicking a hazardous connection might introduce malware
Lock the framework as a feature of a ransomware assault, or uncover private data
Enabling two step verification in emails
PAIN
fears
frustrations
obstacles
Usage of authentic logos
Loss of confidentiality
Money could be stolen through online transaction
Important data would be revealed
GAIN
"wants" / needs
measures of success
obstacles
Private information of authorities can be fetched
Attackers obtain user's money
Gaining priveleged access
Private email domain for personal use

3.2 Ideation & Brainstorming:

Ideation essentially refers to the whole creative process of coming up with and communicating new ideas. Ideation is innovative thinking, typically aimed at solving a problem or providing a more efficient means of doing or accomplishing something.

Brainstorming simply relies on "The best way to have a good idea is to have lots of ideas". Visual idealization will challenge the more logically inclined to think differently. And different thinking facilitates unique, worthwhile ideas.

Ideation is often closely related to the practice of brainstorming, a specific technique that is utilized to generate new ideas. A principal difference between ideation and brainstorming is that ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity.


Brainstorm
& idea prioritization
Shafiqur Rahman
Sathish Kumar
Shiyam Ram
Vijaya Prashanth
Grouping based on URL:
Grouping based on dataset
Machine learning can be used to extract phishing features from a data set
Any classification algorithm can be used for the prediction
Using proper dataset we can achive high accuracy
Prepare an outline on how to proceed the problem
Improper popups leading to various website arises

3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to besolved)	To detect the phishing websites.
2.	Idea / Solution description	To develop a machine learning model with highaccuracy that uses classification algorithm to find whether a website is a phishing website or not.
3.	Novelty / Uniqueness	Pre-processing the dataset and training them by appropriate machine learning classification algorithm
4.	Social Impact / Customer Satisfaction	People are safeguarded from threats and theirinformation are secured.
5.	Business Model (Revenue Model)	The model can be integrated as a service andprovided to the public. With various advancements it will be a good revenue producing model.
6.	Scalability of the Solution	It can support multiple requests for identifyingthe websites.

3.4 Problem Solution Fit:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it solves the customer’s problem. It helps entrepreneurs,marketers and corporate innovators identify behavioral patterns and recognize what would work and why.

Purpose:

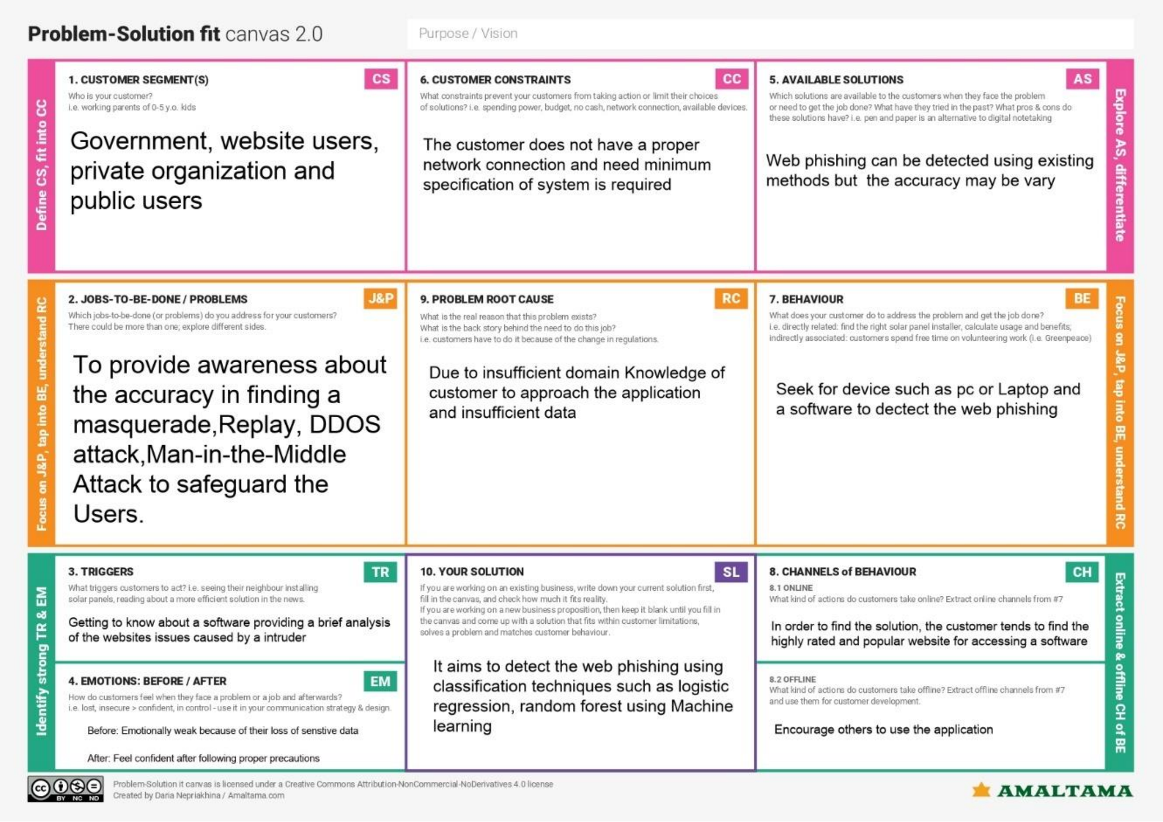
❑ Solve complex problems in a way that fits the state of your customers.

❑ Succeed faster and increase your solution adoption by tapping into existing mediums andchannels of behavior.

❑ Sharpen your communication and marketing strategy with the right triggers and messaging.

❑ Increase touchpoints with your company by finding the right problem-behaviour fit and building trust by solving frequent annoyances, or urgent or costly problems.

❑ Understand the existing situation in order to improve it for your target group.



Problem-Solution fit canvas 2.0

Purpose / Vision

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

Government, website users, private organization and public users

6. CUSTOMER CONSTRAINTS CC

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.

The customer does not have a proper network connection and need minimum specification of system is required

5. AVAILABLE SOLUTIONS AS

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 notetaking

Web phishing can be detected using existing methods but the accuracy may be vary

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers?
There could be more than one; explore different sides.

To provide awareness about the accuracy in finding a masquerade,Replay, DDOS attack,Man-in-the-Middle Attack to safeguard the Users.

9. PROBLEM ROOT CAUSE RC

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.

Due to insufficient domain Knowledge of customer to approach the application and insufficient data

7. BEHAVIOUR BE

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 free time on volunteering work (i.e. Greenpeace)

Seek for device such as pc or Laptop and a software to dectect the web phishing

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

Getting to know about a software providing a brief analysis of the websites issues caused by a intruder

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

Before: Emotionally weak because of their loss of senstive data

After: Feel confident after following proper precautions

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
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.

It aims to detect the web phishing using classification techniques such as logistic regression, random forest using Machine learning

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

In order to find the solution, the customer tends to find the highly rated and popular website for accessing a software

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

Encourage others to use the application

Extract online & offline CH of BE

AMALTAMA

CHAPTER 4

REQUIREMENT ANALYSIS

4.2 Functional Requirements:

FR No.	**Functional Requirement (Epic)**	**Sub Requirement (Story / Sub-Task)**
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIn
FR-2	User Confirmation	Confirmation via EmailConfirmation via OTP
FR-3	Helpdesk	User could be able to get guidance from the customer care
FR-4	Management	Administrator must collect new datasets and keep the model trained

4.2 Non-Functional Requirements:

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	The system must be efficient and easy for the user to execute the tasks.
NFR-2	**Security**	User data protection must be ensured.
NFR-3	**Reliability**	The result produced must be reliable.
NFR-4	**Performance**	The website must be able to produce the results at fast pace.
NFR-5	**Availability**	The website must be available to all users.
NFR-6	**Scalability**	The website should be capable of handling heavy traffic.

CHAPTER 5

PROJECT DESIGN

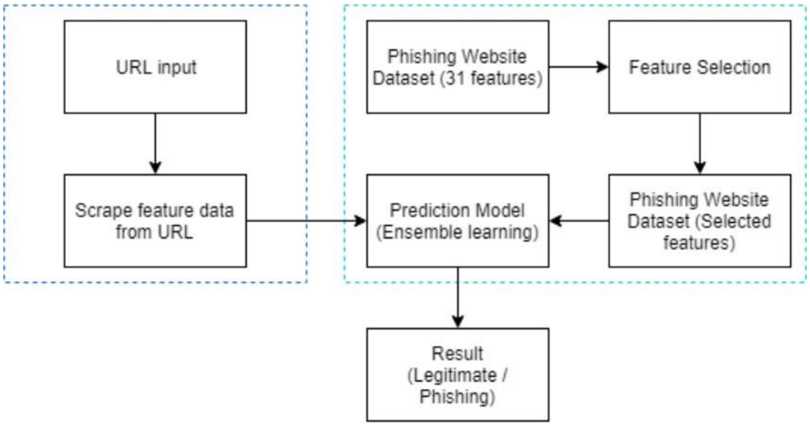
5.1 Data Flow Diagram:

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Itshows how data enters and leaves the system, what changes the information, and where data is stored.

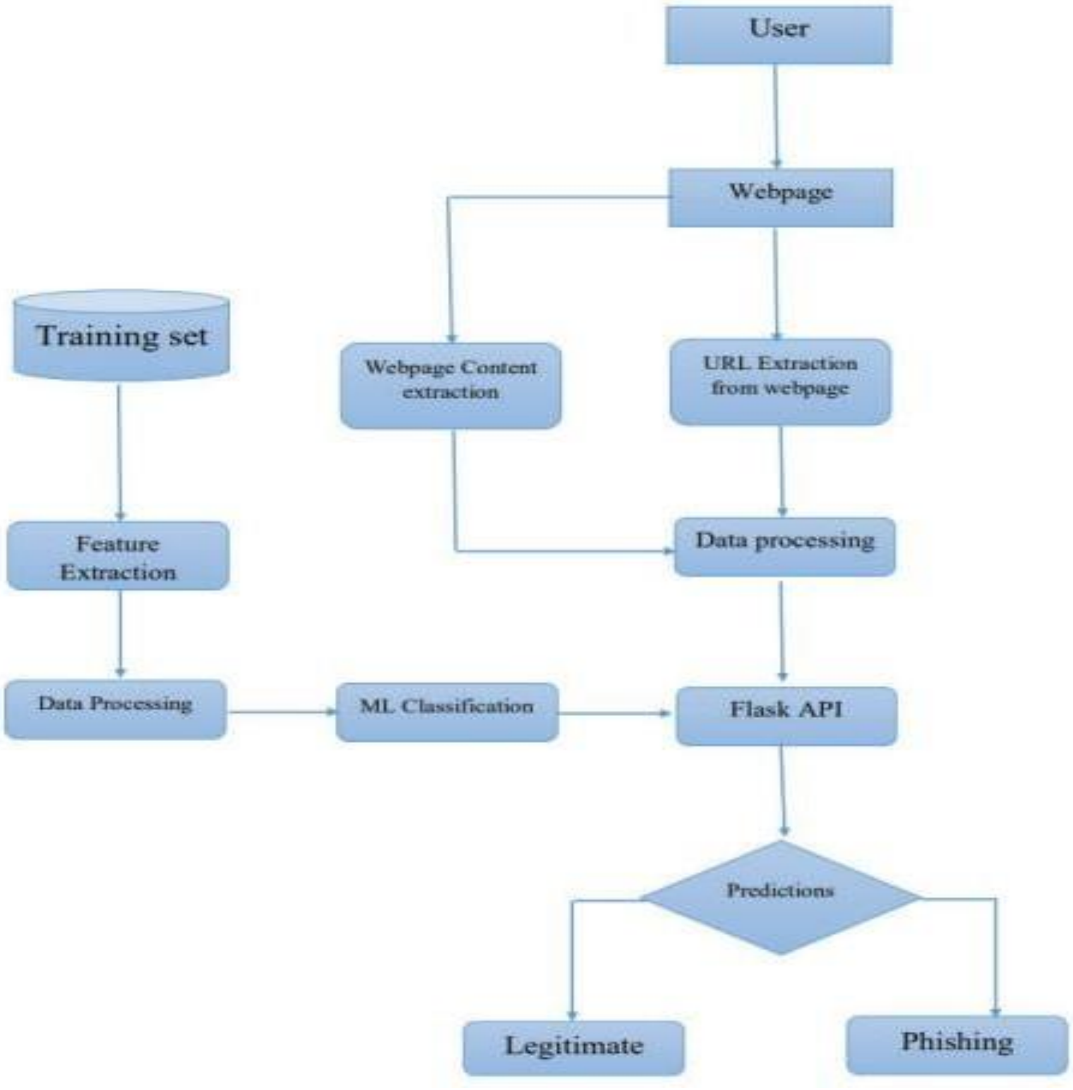
DFD LEVEL 0:

DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled. Its designed to be an at-a-glance view, showing the system as a single high-level process, with its relationship to external entities.

Flow:


URL input
Scrape feature data from URL
Phishing Website Dataset (31 features)
Feature Selection
Prediction Model (Ensemble learning)
Phishing Website Dataset (Selected features)
Result (Legitimate / Phishing)

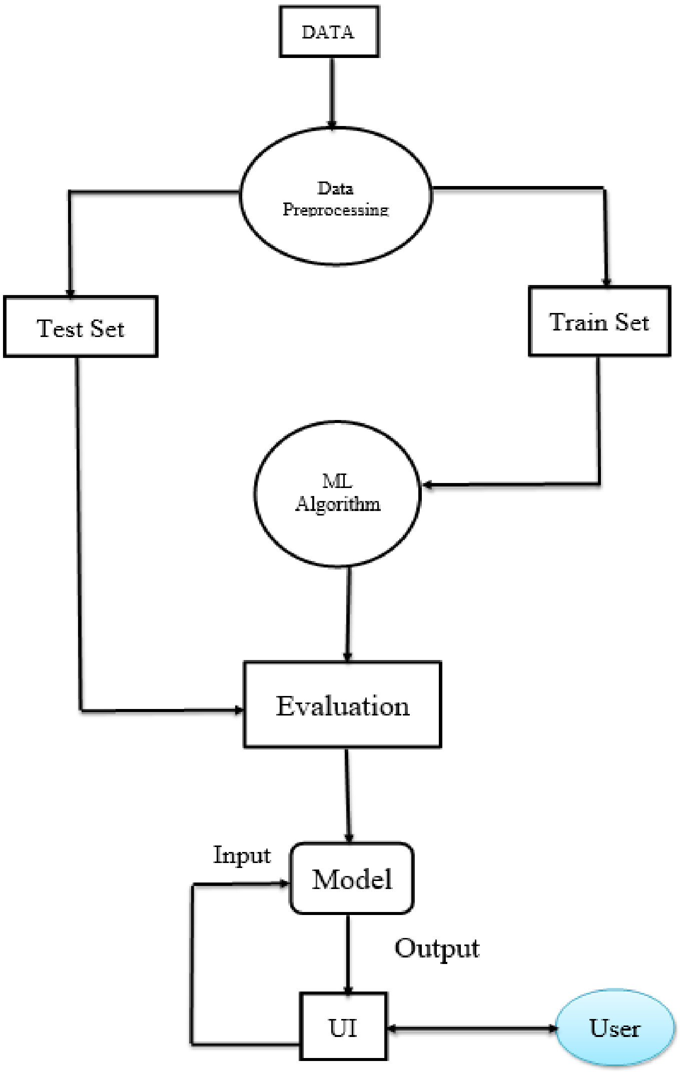
1. User enters the URL and submits.
2. The website checks the URL .
3. The link will be checked with phishing activites.
4. The result of the prediction will be stored.
5. The result will be intimated to the user.



5.2 Solution & Technical Architecture:

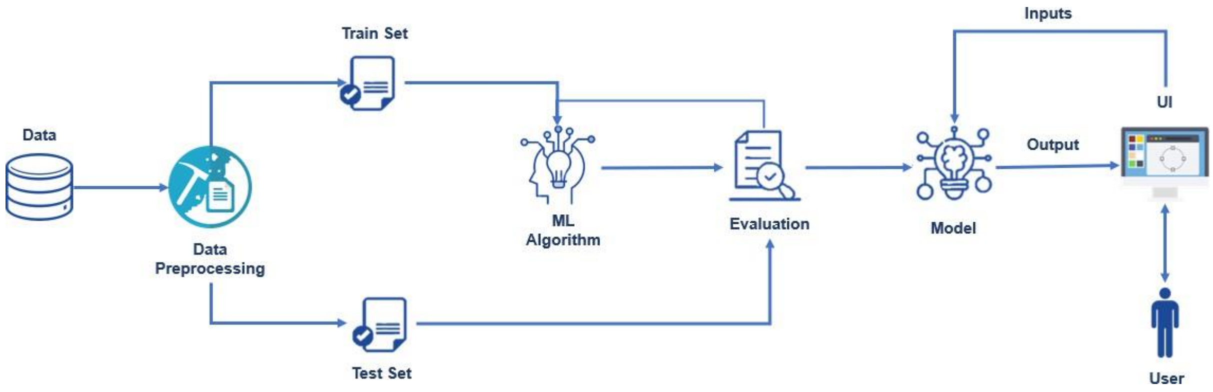
SOLUTION ARCHITECTURE:

Solution architecture defines how those requirements would translate into the way a given software operates. Our solution is to build an efficient and intelligent system to detect phishing sites by applying a machine learning algorithm which implements classification algorithms and techniques to extract the phishing datasets criteria to classify their legitimacy by carefully analyzing and identifying variousfactors that could be used to detect a phishing site.



TECHNICAL ARCHITECTURE:

Technical architecture which is also often referred to as application architecture includes the major components of the system, their relationships, and the contracts that define the interactions between the components. The goal of technical architects is to achieve all the business needs with an application that is optimized for both performance and security.



5.3 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 web 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 Ihave registered for the web application	I can receive confirmation email& click confirm	High	Sprint-1
	Login	USN-3	As a user, I can log into the application byentering email & password	I can login & access my accountwith my registered credentials	High	Sprint-1
	Dashboard	USN-4	As a user, I can access the services and information provided in the dashboard	I can upload the website link and can view the result	High	Sprint-1
Customer (Webuser)	Login	USN-5	As a user, I can log into the web application and access the dashboard	I can login with the same registered credentials and access my account through web application	High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer Care Executive	Help Desk	USN-6	As a user, I can get the guidance from the customer care	I can get help from the customer care for carrying out mytasks	High	Sprint-2
Administrator	Management	USN-7	As an administrator, I can collect new datasets and keep the model trained	I can collect and train the model with new dataset frequently	High	Sprint-2

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation:

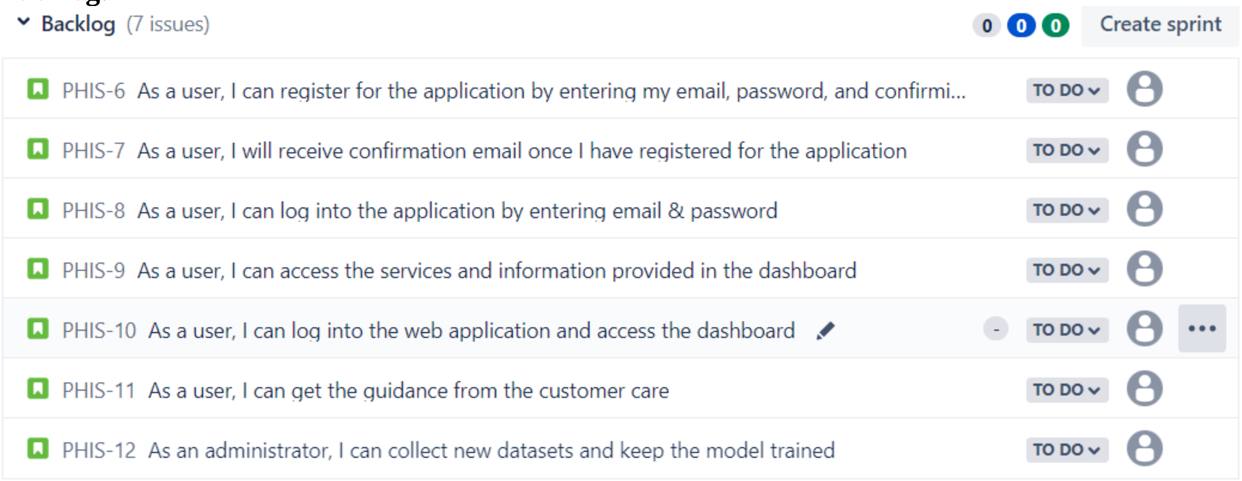
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirmingmy password.	10	High	Shiyam Ram
Sprint-1		USN-2	As a user, I will receive confirmation email onceI have registered for the application	5	High	Shafiqur Rahman
Sprint-1	Login	USN-3	As a user, I can log into the application byentering email & password	5	High	Vijaya Prashanth
Sprint-2	Dashboard	USN-4	As a user, I can access the services andinformation provided in the dashboard	15	High	Sathish Kumar
Sprint-2		USN-5	As a user, I can log into the web application and access the dashboard	5	High	Sathish Kumar
Sprint-3	Help Desk	USN-6	As a user, I can get the guidance from thecustomer care	10	High	Vijaya Prashanth
Sprint-4	Management	USN-7	As an administrator, I can collect new datasets and keep the model trained	10	High	Shafiqur Rahman

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		
Sprint-3	10	6 Days	07 Nov 2022	12 Nov 2022		
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022		

6.3 Reports from JIRA:

Backlog:



Backlog (7 issues) 0 0 0 Create sprint

PHIS-6 As a user, I can register for the application by entering my email, password, and confirmi... TO DO

PHIS-7 As a user, I will receive confirmation email once I have registered for the application TO DO

PHIS-8 As a user, I can log into the application by entering email & password TO DO

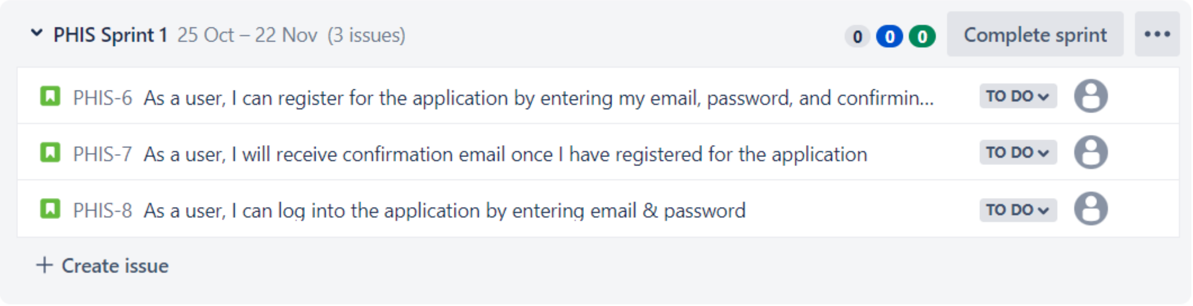
PHIS-9 As a user, I can access the services and information provided in the dashboard TO DO

PHIS-10 As a user, I can log into the web application and access the dashboard - TO DO

PHIS-11 As a user, I can get the guidance from the customer care TO DO

PHIS-12 As an administrator, I can collect new datasets and keep the model trained TO DO

Sprint 1:



PHIS Sprint 1 25 Oct – 22 Nov (3 issues) 0 0 0 Complete sprint

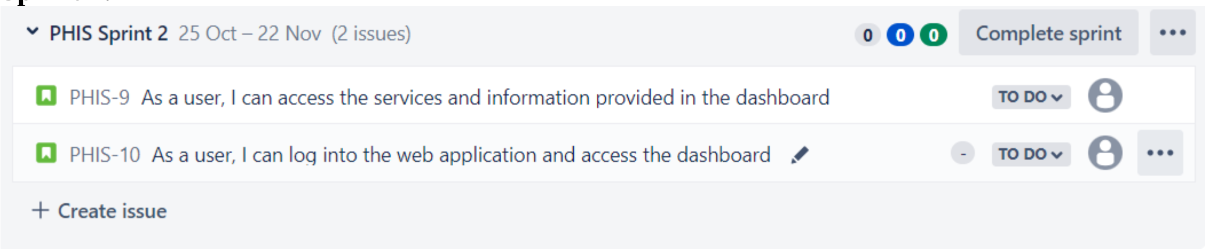
PHIS-6 As a user, I can register for the application by entering my email, password, and confirmin... TO DO

PHIS-7 As a user, I will receive confirmation email once I have registered for the application TO DO

PHIS-8 As a user, I can log into the application by entering email & password TO DO

+ Create issue

Sprint 2:



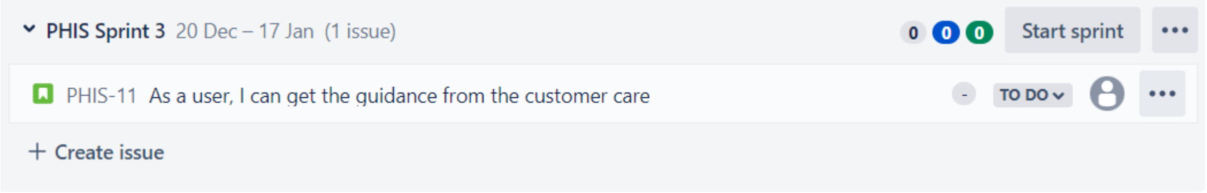
PHIS Sprint 2 25 Oct – 22 Nov (2 issues) 0 0 0 Complete sprint

PHIS-9 As a user, I can access the services and information provided in the dashboard TO DO

PHIS-10 As a user, I can log into the web application and access the dashboard - TO DO

+ Create issue

Sprint 3:

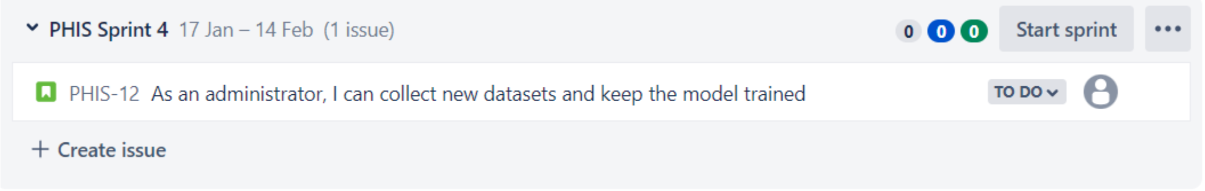


PHIS Sprint 3 20 Dec – 17 Jan (1 issue) 0 0 0 Start sprint

PHIS-11 As a user, I can get the guidance from the customer care - TO DO

+ Create issue

Sprint 4:

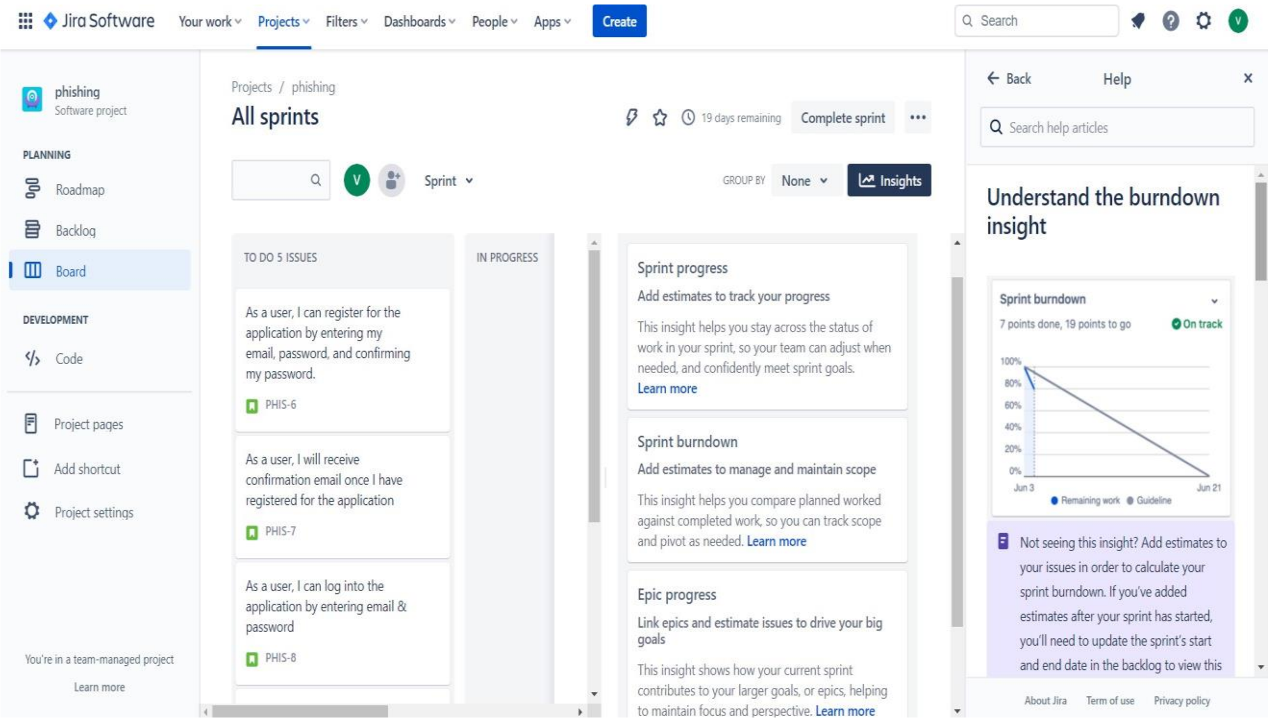


PHIS Sprint 4 17 Jan – 14 Feb (1 issue) 0 0 0 Start sprint

PHIS-12 As an administrator, I can collect new datasets and keep the model trained TO DO

+ Create issue

Insights:



phishing
Software project

PLANNING

Roadmap

Backlog

Board

DEVELOPMENT

Code

Project pages

Add shortcut

Project settings

You're in a team-managed project

Learn more

Projects / phishing

All sprints

19 days remaining

Complete sprint

TO DO 5 ISSUES

As a user, I can register for the application by entering my email, password, and confirming my password.

PHIS-6

As a user, I will receive confirmation email once I have registered for the application

PHIS-7

As a user, I can log into the application by entering email & password

PHIS-8

IN PROGRESS

Sprint progress

Add estimates to track your progress

This insight helps you stay across the status of work in your sprint, so your team can adjust when needed, and confidently meet sprint goals. Learn more

Sprint burndown

Add estimates to manage and maintain scope

This insight helps you compare planned worked against completed work, so you can track scope and pivot as needed. Learn more

Epic progress

Link epics and estimate issues to drive your big goals

This insight shows how your current sprint contributes to your larger goals, or epics, helping to maintain focus and perspective. Learn more

Back

Help

Search help articles

Understand the burndown insight

Sprint burndown

7 points done, 19 points to go

On track

Not seeing this insight? Add estimates to your issues in order to calculate your sprint burndown. If you've added estimates after your sprint has started, you'll need to update the sprint's start and end date in the backlog to view this

About Jira Term of use Privacy policy

CHAPTER 7

CODING & SOLUTIONING

7.1 Feature 1 – Classification of URL:

The primary feature of this project is to classify the given URL as phishing or safe. Various classification algorithms are used to achieve this.

7.1.1 Methodology:

7.1.1.1 Data Collection:

URL features of legitimate websites and phishing websites were collected. The data set consists of total 11,055 URLs which include 6,157 legitimate URLs and 4,898 phishing URLs. Legitimate URLs are labelled as “1” and phishing URLs are labelled as “-1”. The features that are present in the data set include:

- IP Address in URL
- Length of URL
- Using URL Shortening Services
- "@" Symbol in URL
- Redirection "//" in URL
- Prefix or Suffix "-" in Domain
- Having Sub Domain
- Length of Domain Registration
- Favicon
- Port Number
- HTTPS Token
- Request URL
- URL of Anchor
- Links in Tags
- SFH
- Email Submission
- Abnormal URL
- Status Bar Customization (on mouse over)
- Disabling Right Click
- Presence of Popup Window
- IFrame Redirection
- Age of Domain
- DNS Record
- Web Traffic
- Page Rank
- Google Index
- Links pointing to the page
- Statistical Report
- Result

```
import os, types
import pandas as pd
```

```
from botocore.client import Config
import ibm_boto3
def __iter__(self): return 0
# The following code accesses a file in your IBM Cloud Object
Storage. It includes your credentials.
# You might want to remove those credentials before you share the
notebook.
cos_client = ibm_boto3.client(service_name='s3',
 ibm_api_key_id=",
 ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
 config=Config(signature_version='oauth'),
 endpoint_url='https://s3.private.us.cloud-object   storage.appdomain.cloud')
bucket = 'webphishingdetection-donotdelete-pr-icmjtvktnzli2s'
object_key = 'dataset_website.csv'
body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like
object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType(
__iter__, body )
data0 = pd.read_csv(body)
data0.head()
```

7.1.1.2 Model building:

From the dataset above, it is clear that this is a supervised machine learning task. There are two major types of supervised machine learning problems, called classification and regression.
This data set comes under classification problem, as the input URL is classified as phishing (-1) or legitimate (1). The supervised machine learning models (classification) considered to train the dataset in this notebook is Logistic Regression

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=0)
lr = LogisticRegression()
lr.fit(x_train,y_train)
y_pred1 = lr.predict(x_test)
log_reg = accuracy_score(y_test,y_pred1)
```

CHAPTER 8
TESTING

8.1 Test Cases:

Test case ID	Component	Test Scenario	Pre-Requisite	Steps To Execute	Expected Result	Actual Result	Status
SignupPage_TC_OO1	Signup Page	Verify the user is able to see the page.	Run the flask app in local host	1.Open our phishing website 2.Click signup.	Page is diplsayed.	Working as expected	Pass
SignupPage_TC_OO2	Signup Page	Verify the user can enter details and submit	Run the flask app in local host	1.Enter details and submit.	Registration succesfull and redirects to home page.	Working as expected	Pass
SignupPage_TC_OO2	Signup Page	Verify the email is sent to User	Run the flask app in local host	1.Enter details and submit.	Email is sent to the user.	Working as expected	Pass
LoginPage_TC_OO1	Login Page	Verify the UI elements in Login	Run the flask app in local host	1.Open our phishing website 2.Click login	Page is diplsayed.	Working as expected	Pass
LoginPage_TC_OO2	Login Page	Verify the user can enter details and login	Run the flask app in local host	1.Open our phishing website 2.Click login	Redirects to my account page.	Working as expected	Pass

8.2 User Acceptance Testing:

Defect Analysis

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

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

Test Case Analysis

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

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

CHAPTER 9

RESULTS

9.1 Performance metrics:

The median efficiency is used to assess each categorization model's effectiveness. The final item will appear in the way it was envisioned. Graphical representations are used to depict information during classification. The percentage of predictions made using the testing dataset is used to gauge accuracy. By dividing the entire number of forecasts even by properly predicted estimates, it is simple to calculate. The difference between actual and anticipated output is used to calculate accuracy.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where TP = True Positives, TN = True Negatives, FN = False Negatives and FP = False Positives.

CHAPTER 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- This system can be used by many E-commerce or other websites in order to have good customer relationship.
- URL feeds are verified for legitimacy, the system detects phishing sites by applying a machine learning algorithm which implements classification algorithms and techniques to extract the phishing datasets criteria to classify their legitimacywhich in turn helps the customers to eliminate the risks of cyber threat and protect their valuable corporate or personal data.
- Clients can report to administrator which helps them to ask their questions significantly improving their experience on our site
- Requiring low resources on host machine

DISADVANTAGES:

- If Internet connection fails, this system won't work.
- Needs continuous feed of URL links in order to achieve a generalized model
- All websites related data will be stored in one place
- Effective only when minimal FP rates are required.

CHAPTER 11

CONCLUSION

Phishing is growing continuously irrespective of intelligence security development, there is definitely need of special care toward safeguarding of people being cheated. The accessibility of defect of an attacker of phishing device is additionally imperative to built the extent of anticipating phishing sites. Phishing detection is now an area of great interest among the researchers due to its significance in protecting privacy and providing security. Phishing detection schemes perform better than phishing prevention and user training solutions because they do not require changes in authentication platforms and do not rely on the user's ability to detect phishing. There are many methods to perform phishing detection. Our system aims to enhance the detection method to detect phishing websites using machine learning technology. The proposed algorithm has also been compared with work on different dataset with similar algorithm and the results show that the proposed model achieves considerably better accuracy as compared to works.

CHAPTER 12

FUTURE SCOPE

Further enhancement of this work could be use of more advanced algorithms with 10-fold cross validation. Feature selection method can also be varied to see the effect on the varies parameters. A combination or hybrid machine learning algorithm can also be implemented to improve success rate and minimize false rate. We can also use a combination of any two or more classifiers to get maximum accuracy. In future if we get structured dataset of phishing we can perform phishing detection much more faster than any other technique. In future we can use a combination of any other two or more classifier to get maximum accuracy. In particular, we extract features from URLs and pass it through the various classifiers.

We plan to explore various phishing techniques which use Network based features, Content based features, Webpage based features and HTML and JavaScript features of web pages which will improve the performance of the system. In particular, we extract features from URLs and pass it through the various classifiers. In future, hybrid technology will be implemented to detect phishing websites more accurately.

CHAPTER 13

APPENDIX

13.1 Source Code:

App.py

```
import os
from flask_sqlalchemy import SQLAlchemy
from flask_migrate import Migrate
from werkzeug.security import generate_password_hash,check_password_hash
from flask import Flask,render_template,url_for,flash,redirect,request
from flask_login import login_user, current_user, logout_user,
login_required,LoginManager,UserMixin
import smtplib
import pickle

#importing the inputscript file used to analyze the URL
import inputScript
model = pickle.load(open('Phishing_website.pkl','rb'))

app =Flask(__name__)
app.config['SECRET_KEY'] ='mysecret'

basedir =os.path.abspath(os.path.dirname(__file__))
app.config['SQLALCHEMY_DATABASE_URI']='sqlite:///'+os.path.join(basedir,'
data.sqlite')
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] =False

db=SQLAlchemy(app)
Migrate(app,db)

login_manager = LoginManager()

login_manager.init_app(app)
login_manager.login_view = 'users.login'

@login_manager.user_loader
def load_user(user_id):
    return User.query.get(user_id)


class User(db.Model,UserMixin):

    __tablename__ = 'users'

    id = db.Column(db.Integer,primary_key=True)
    email = db.Column(db.String(64),unique=True,index=True)
    name = db.Column(db.String(64),index=True)
    password_hash = db.Column(db.String(128))

    def __init__(self,email,name,password):
        self.email = email
        self.name = name
        self.password_hash = generate_password_hash(password)

    def check_password(self,password):
        return check_password_hash(self.password_hash,password)


    def __repr__(self):
        return f"name {self.name}"


    def check_email(self,field):
        if User.query.filter_by(email=field.data).first():
            flash('Your email has been registered already!')

class Phishing(db.Model):
    __tablename__ = 'url_info'

    id = db.Column(db.Integer,primary_key=True)
    url = db.Column(db.String(64),index=True)
    email = db.Column(db.String(64),unique=False,index=True)
    result = db.Column(db.String(64),index=True)

    def __init__(self,url,email,result):
        self.result = result
        self.email = email
        self.url=url

    def __repr__(self):
        return f"url {self.url}"

@app.route('/')
def home():
    return render_template('home.html')

@app.route('/check',methods=['GET','POST'])
def check():
    if request.method =='POST':
        url = request.form.get("web-url")
        check = inputScript.main(url)
        prediction = model.predict(check)
        output = prediction[0]
        if(output == 1):
            message = "You are Safe. It is a Legitimate Website."
            res = 1
        else:
            message = "Alert! It is a malicious Website."
            res = 0
        if current_user.is_authenticated:
            with open("file.txt") as file:
                email = file.read()
            db.create_all()
            result = message
            web_check = Phishing(url=url,email=email,result=result)
            db.session.add(web_check)
            db.session.commit()
        return render_template('check.html', message = message, url =
url, res = res)
    else:
        return render_template('check.html')

@app.route('/signup',methods=['GET','POST'])
def register():
    if request.method =='POST':
        db.create_all()
        Name = request.form.get("name")
        Email = request.form.get("email")
        password = request.form.get("password")
        conf_password = request.form.get("cpassword")

        if password ==conf_password:
            user = User(email=Email,name=Name,password=password)
            db.session.add(user)
            db.session.commit()
            send_email(name = Name, email=Email)
            with open("file.txt", "w") as file:
                file.write(Email)

            user = User.query.filter_by(email=Email).first()
            if user.check_password(password) and user is not None:
                login_user(user)
                flash('Login Successful!')
                return redirect(url_for('home'))

        else:
            flash("Passwords do not match")
            return render_template("signup.html")
    else:
        return render_template("signup.html")

def send_email(name, email):
    sender = 'cryptrix22@gmail.com'
    password = 'wajwjomyaockwokc'
    receiver = email

    session = smtplib.SMTP('smtp.gmail.com', 587)

    session.starttls()

    session.login(sender, password)

    text = f'''
    Hello {name},
        Thank you for registering.
        You have successfully created account with BeSafe.
        Browse the internet securely by finding Phishing Websites.

    Regards,
    BeSafe
    '''
    session.sendmail(sender, receiver, text)
    session.quit()
```

```
# login
@app.route('/login',methods=['GET','POST'])
def login():

    if request.method =='POST':
        Email = request.form.get("email")
        password = request.form.get("password")
        user = User.query.filter_by(email=Email).first()

        if user is not None and user.check_password(password) :
            # flash('Log in Success!')
            login_user(user)
            with open("file.txt", "w") as file:
                file.write(Email)
            flash('Login Successful!')
            return redirect(url_for('account'))

        else:
            flash('Username or Password is incorrect')
            return render_template("login.html")
    else:
        return render_template('login.html')

@app.route('/account')
@login_required
def account():
    with open("file.txt") as file:
        Email = file.read()
    user = User.query.filter_by(email=Email).first()
    searches = Phishing.query.filter_by(email=Email).all()
    phishing_count = len(Phishing.query.filter_by(email=Email,
result='Alert! It is a malicious Website.').all())
    legitimate_count = len(Phishing.query.filter_by(email=Email, result =
'You are Safe. It is a Legitimate Website.').all())
    return render_template('account.html',all_search =
searches,email=Email,name=user.name,length = len(searches),p_count =
phishing_count,l_count = legitimate_count)

# logout
@app.route("/logout")
@login_required
def logout():
    logout_user()
    flash('Logged out Successfully')
    return redirect(url_for("home"))

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

account.html

```
{% extends "boilerplate.html" %}
{% block content %}
<div class="row">
  <div class="col-10 offset-1 bg-white shadow-sm p-3">
```

```
    <h1 class="fs-4 text-primary">Welcome {{ name }}</h1>
    <h2 class="fs-5 text-secondary">My Account</h2>
    <div class="row">
      <div class="col-2">
        <div class="">Name</div>
        <div class="">Email</div>
      </div>
      <div class="col-4">
        <div class="">{{ name }}</div>
        <div class="">{{email}}</div>
      </div>
      <div class="col-2">
        <div class="d-flex align-items-center flex-column">
          <div class="bg-primary rounded-circle px-3 py-2" style="width:
fit-content;">
            <div class="fs-4">{{length}}</div>
          </div>
          <span class="fs-5">URL Checked</span>
        </div>
      </div>
      <div class="col-2">
        <div class="d-flex align-items-center flex-column">
          <div class="bg-info rounded-circle px-3 py-2" style="width:
fit-content;">
            <div class="fs-4">{{l_count}}</div>
          </div>
          <span class="fs-5">Legitimate</span>
        </div>
      </div>
      <div class="col-2">
        <div class="d-flex align-items-center flex-column">
          <div class="bg-danger rounded-circle px-3 py-2" style="width:
fit-content;">
            <div class="fs-4">{{p_count}}</div>
          </div>
          <span class="fs-5">Phishing</span>
        </div>
      </div>
    </div>
  </div>
</div>
<div class="row mt-3">
  <div class="col-10 offset-1 bg-white shadow-sm p-3">
    <h2 class="fs-5 text-secondary">Search History</h2>
    <table class="table table-striped">
      <thead>
        <th>S No.</th>
        <th>Website URL</th>
        <th>Result</th>
      </thead>
      <tbody>
        {% for index in range(length) %}
        <tr>
          <td>{{index + 1}}</td>
          <td>{{all_search[index].url}}</td>
          <td>{{all_search[index].result}}</td>
        </tr>
        {% endfor %}
      </tbody>
    </table>
  </div>
</div>
</div>
{% endblock %}
```

Boilerplate.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>BeSafe</title>
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min
.css" rel="stylesheet"
        integrity="sha384-
1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">
</head>

<body class="bg-light d-flex flex-column vh-100">

    <nav class="navbar navbar-expand-lg navbar-light shadow-sm bg-white
sticky-top border-bottom border-primary">
        <div class="container">
            <a class="navbar-brand text-primary" href="/">BeSafe</a>
            <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target="#navbarNavAltMarkup"
                aria-controls="navbarNavAltMarkup" aria-expanded="false"
aria-label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse"
id="navbarNavAltMarkup">
                <div class="navbar-nav">
                    <a class="nav-link text-dark" aria-current="page"
href="/">Home</a>
                    <a class="nav-link text-dark" aria-current="page"
href="{{url_for('check')}}">Check URL</a>
                </div>
                <div class="navbar-nav ms-auto">
                    {% if current_user.is_authenticated %}
                    <a class="nav-link text-dark"
href="{{url_for('account')}}">My Account</a>
                    <a class="nav-link text-dark"
href="{{url_for('logout')}}">Logout</a>
                    {% else %}
                    <a class="nav-link text-dark"
```

```
href="{{url_for('login')}}">Login</a>
                    <a class="nav-link text-dark"
href="{{url_for('register')}}">Signup</a>
                    {% endif %}

                </div>
            </div>
        </div>
    </nav>
    <main class="container my-3">

        {% for message in get_flashed_messages() %}
        <div class="alert alert-success alert-dismissible fade show"
role="alert">
            {{ message }}
            <button type="button" class="btn-close" data-bs-
dismiss="alert" aria-label="Close"></button>
          </div>
        {% endfor %}

        {% block content %}

        {% endblock %}

    </main>

    <footer class="footer mt-auto">
        <div class="text-white" style="background-color: #0255CF;">
            <div class="container d-flex p-3">
                <div class="mx-auto">&copy; 2022 BeSafe. All Rights
Reserved</div>
            </div>
        </div>
    </footer>

    <script src="{{url_for('static',
filename='validateForm.js')}}"></script>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundl
e.min.js"
        integrity="sha384-
ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
crossorigin="anonymous">
        </script>

</body>

</html>
```

Check.html

```
<!DOCTYPE html>
<html lang="en">

<head>
```

```
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>BeSafe</title>
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min
.css" rel="stylesheet"
        integrity="sha384-
1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">
</head>

<body class="bg-light d-flex flex-column vh-100">

    <nav class="navbar navbar-expand-lg navbar-light shadow-sm bg-white
sticky-top border-bottom border-primary">
        <div class="container">
            <a class="navbar-brand text-primary" href="/">BeSafe</a>
            <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target="#navbarNavAltMarkup"
                aria-controls="navbarNavAltMarkup" aria-expanded="false"
aria-label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse"
id="navbarNavAltMarkup">
                <div class="navbar-nav">
                    <a class="nav-link text-dark" aria-current="page"
href="/">Home</a>
                    <a class="nav-link text-dark" aria-current="page"
href="{{url_for('check')}}">Check URL</a>
                </div>
                <div class="navbar-nav ms-auto">
                    {% if current_user.is_authenticated %}
                    <a class="nav-link text-dark"
href="{{url_for('account')}}">My Account</a>
                    <a class="nav-link text-dark"
href="{{url_for('logout')}}">Logout</a>
                    {% else %}
                    <a class="nav-link text-dark"
href="{{url_for('login')}}">Login</a>
                    <a class="nav-link text-dark"
href="{{url_for('register')}}">Signup</a>
                    {% endif %}

                </div>
            </div>
        </div>
    </nav>
    <main class="container my-3">

        {% for message in get_flashed_messages() %}
        <div class="alert alert-success alert-dismissible fade show"
role="alert">
            {{ message }}
            <button type="button" class="btn-close" data-bs-
dismiss="alert" aria-label="Close"></button>
          </div>
        {% endfor %}

        {% block content %}

        {% endblock %}

    </main>

    <footer class="footer mt-auto">
        <div class="text-white" style="background-color: #0255CF;">
            <div class="container d-flex p-3">
                <div class="mx-auto">&copy; 2022 BeSafe. All Rights
Reserved</div>
            </div>
        </div>
    </footer>

    <script src="{{url_for('static',
filename='validateForm.js')}}"></script>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundl
e.min.js"
        integrity="sha384-
ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
crossorigin="anonymous">
        </script>

</body>

</html>
```

Home.html

```
{% extends "boilerplate.html" %}
{% block content %}
<div class="row align-items-center bg-white shadow-sm ">
    <div class="col-6 mb-5 mt-5 ps-4">
        <h2 class="fs-4 text-primary">What is Web Phishing ?</h2>
        <p class="lead">Phishing is a type of social engineering attack
often used to steal user data,
            including login credentials and credit card numbers.</p>
        <a href="{{url_for('check')}}" class="btn btn-primary rounded-
pill">Check URL</a>
    </div>
    <div class="col-6 px-0">
        <img src="https://media.kasperskycontenthub.com/wp-
content/uploads/sites/43/2018/12/07085924/abstract-personal-data-
800x450.jpg"
            class="img-fluid" alt="">
    </div>
```

```
</div>
<div class="row align-items-center bg-primary shadow-sm text-white mt-2">
    <div class="col mb-5 mt-5 px-4">
        <h2 class="fs-4">How can we detect ?</h2>
        <ul>
            <li>Phishing is one of the most frequent forms of cyber
crime, but despite how much we think
                we know about these scams, they still catch us out all
too often.</li>
            <li><span class="fw-bold">The message is sent from a public
email domain.</span> No
                legitimate organisation will send emails from an address
that ends '@gmail.com'.</li>
            <li><span class="fw-bold">The domain name is misspelt.</span>
There’s another clue hidden in
                domain names that provides a strong indication of
phishing scams -- unfortunately, it
                complicates our previous clue.</li>
            <li><span class="fw-bold">The email is poorly written.</span>
You can often tell if an email
                is a scam if it contains poor spelling and grammar.</li>
        </ul>

    </div>

</div>
<div class="row align-items-center bg-white shadow-sm mt-2">
    <div class="col mb-5 mt-5 px-4">
        <h2 class="fs-4 text-secondary">How our model works</h2>
        <p class="fs-6">In order to detect and predict 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 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.</p>
        <img
src="https://lh4.googleusercontent.com/oohEvy6ZvTTr7oKH1fL_lPeOkdiDQJbtEW
cxIe4_CnFcjj5jnFB3tib6sN627fFztSzTSAVjvepuUVNYQo4913L0IVN_VCz7ItxnlJWtURh
tg8xan8wTBT8GxMJ3iN1yquYJTi1R"
            class="img-fluid ms-4" alt="">
    </div>
</div>
{% endblock %}
```

Login.html

```
{% extends "boilerplate.html" %}
{% block content %}
<div class="row ">
    <div class="col-10 offset-1">

        <div class="bg-white shadow-sm p-3">
            <div class="row">

                <div class="col-7">
                    <h1 class="fs-3 text-primary">Hello! <br> Welcome
back </h1>
                </div>
                <div class="col-5">
                    <h1>Login</h1>
                    <form action="{{url_for('login')}}" method="post"
class="needs-validation d-flex flex-column"
                        novalidate>
                        <div class="mb-3">
                            <label for="email" class="form-
label">Email</label>
                            <input type="email" class="form-control"
name="email" id="email" required>
                        </div>
                        <div class="mb-3">
                            <label for="password" class="form-
label">Password</label>
                            <input type="password" class="form-control"
name="password" id="password" required>
                        </div>
                        <button class="btn btn-primary ms-auto w-100
rounded-pill ">Login</button>
                        <p class="mt-2 align-self-center">Don't have an
Account ? <span><a
                                    href="{{url_for('register')}}"
class="text-muted">Signup</a></span></p>
                    </form>
                </div>
            </div>
        </div>

    </div>
</div>
</div>

{% endblock %}
```

Signup.html

```
{% extends "boilerplate.html" %}
{% block content %}

<div class="row">
    <div class="col-10 offset-1">
        <div class="bg-white shadow-sm p-3">
            <div class="row">
                <div class="col-7">
                    <h1 class="fs-3 text-primary">Welcome to <span
```

```
class="fw-bold">Web Phishing</span></h1>
                    <h2 class="fs-5">Detecting Phishing websites</h2>
                </div>
                <div class="col-5">
                    <h1>Signup</h1>
                    <form action="{{url_for('register')}}" method="post"
class="needs-validation d-flex flex-column"
                        novalidate>
                        <div class="mb-3">
                            <label for="email" class="form-
label">Email</label>
                            <input type="email" class="form-control"
name="email" id="email" required>
                        </div>
                        <div class="mb-3">
                            <label for="name" class="form-
label">Name</label>
                            <input type="text" class="form-control"
name="name" id="name" required>
                        </div>
                        <div class="mb-3">
                            <label for="password" class="form-
label">Password</label>
                            <input type="password" class="form-control"
name="password" id="password" required>
                        </div>
                        <div class="mb-3">
                            <div class="mb-3">
                                <label for="cpassword" class="form-
label">Confirm Password</label>
                                <input type="password" class="form-
control" name="cpassword" id="cpassword" required>
                            </div>
                            <button class="btn btn-primary rounded-pill
w-100">Signup</button>
                            <p class="mt-2 align-self-center">Already
have an account ? <span><a
                                        href="{{url_for('login')}}"
class="text-muted">Login</a></span></p>
                    </form>
                </div>
            </div>
        </div>
    </div>
</div>
{% endblock %}
```

Validatefrom.js

```
// Example starter JavaScript for disabling form submissions if there are
invalid fields
(function () {
    'use strict'

    // Fetch all the forms we want to apply custom Bootstrap validation
styles to
```

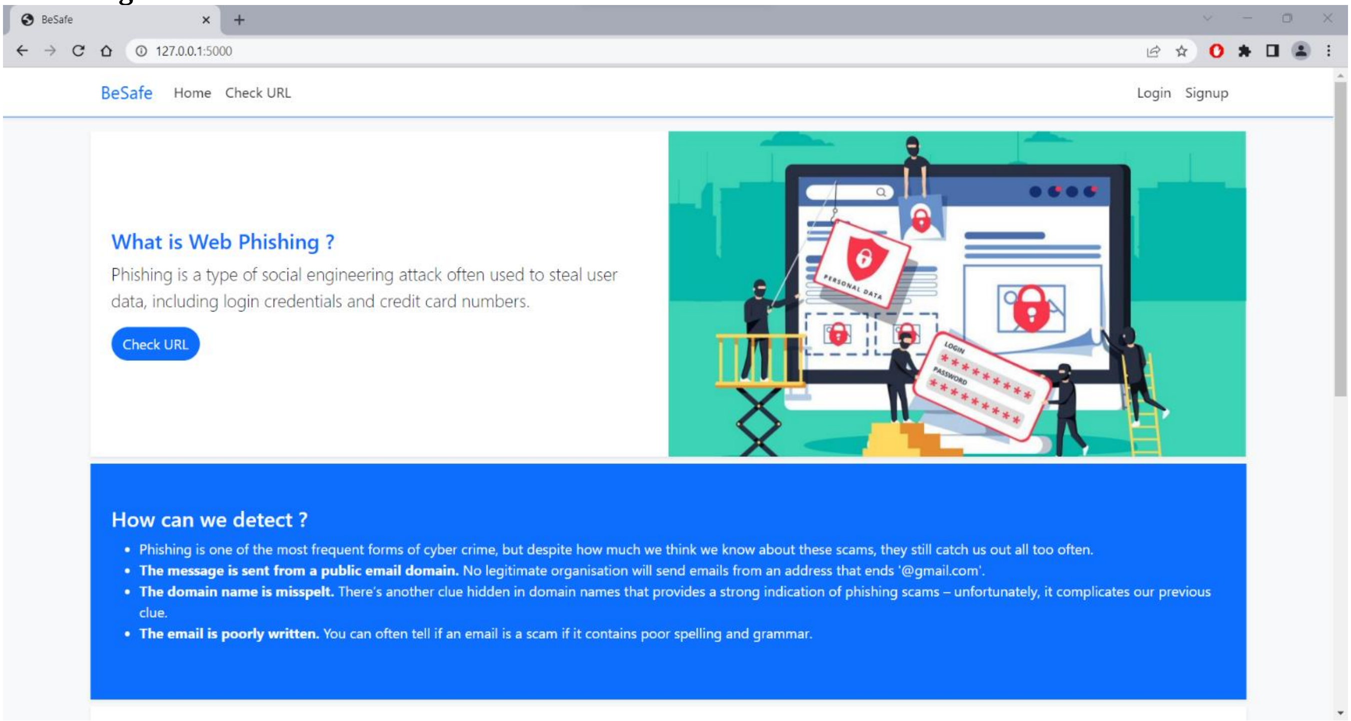
```
    var forms = document.querySelectorAll('.needs-validation')

    // Loop over them and prevent submission
    Array.prototype.slice.call(forms)
      .forEach(function (form) {
        form.addEventListener('submit', function (event) {
          if (!form.checkValidity()) {
            event.preventDefault()
            event.stopPropagation()
          }

          form.classList.add('was-validated')
        }, false)
      })
  })()
```

SCREENSHOTS

Home Page

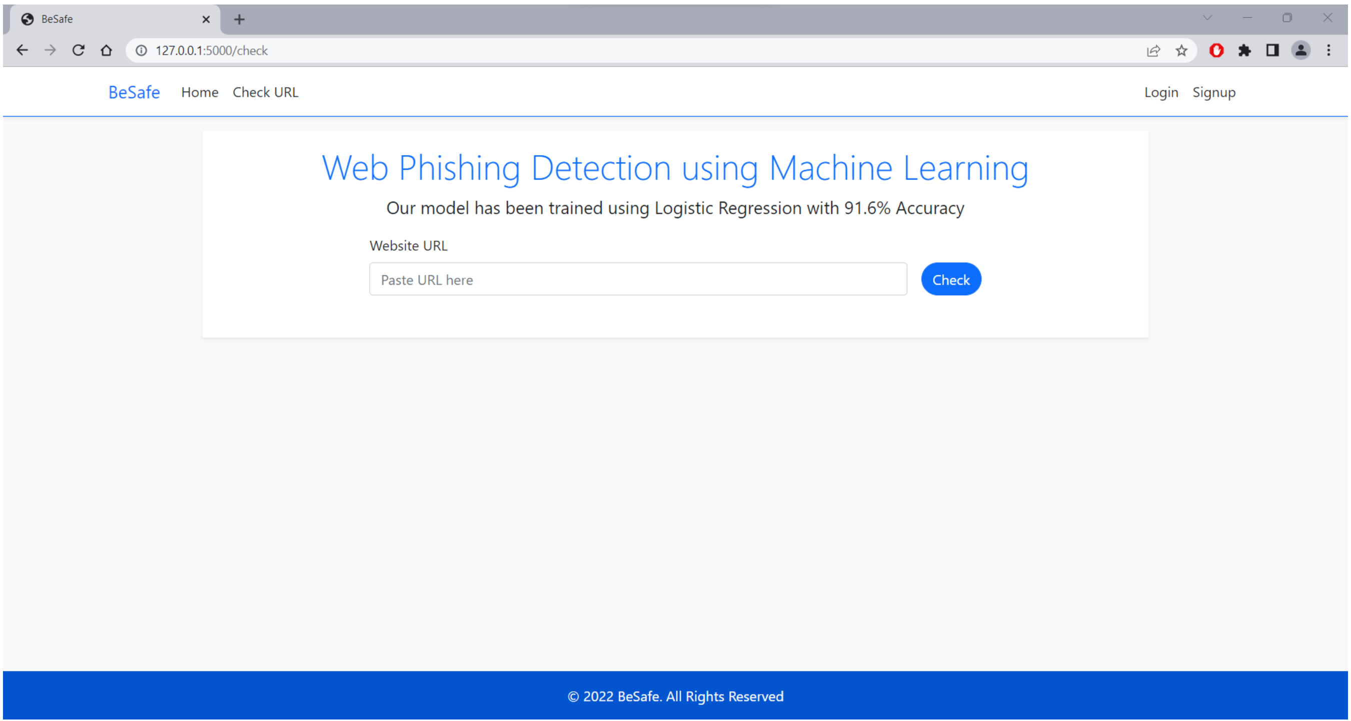


What is Web Phishing ?

Phishing is a type of social engineering attack often used to steal user data, including login credentials and credit card numbers.

How can we detect ?

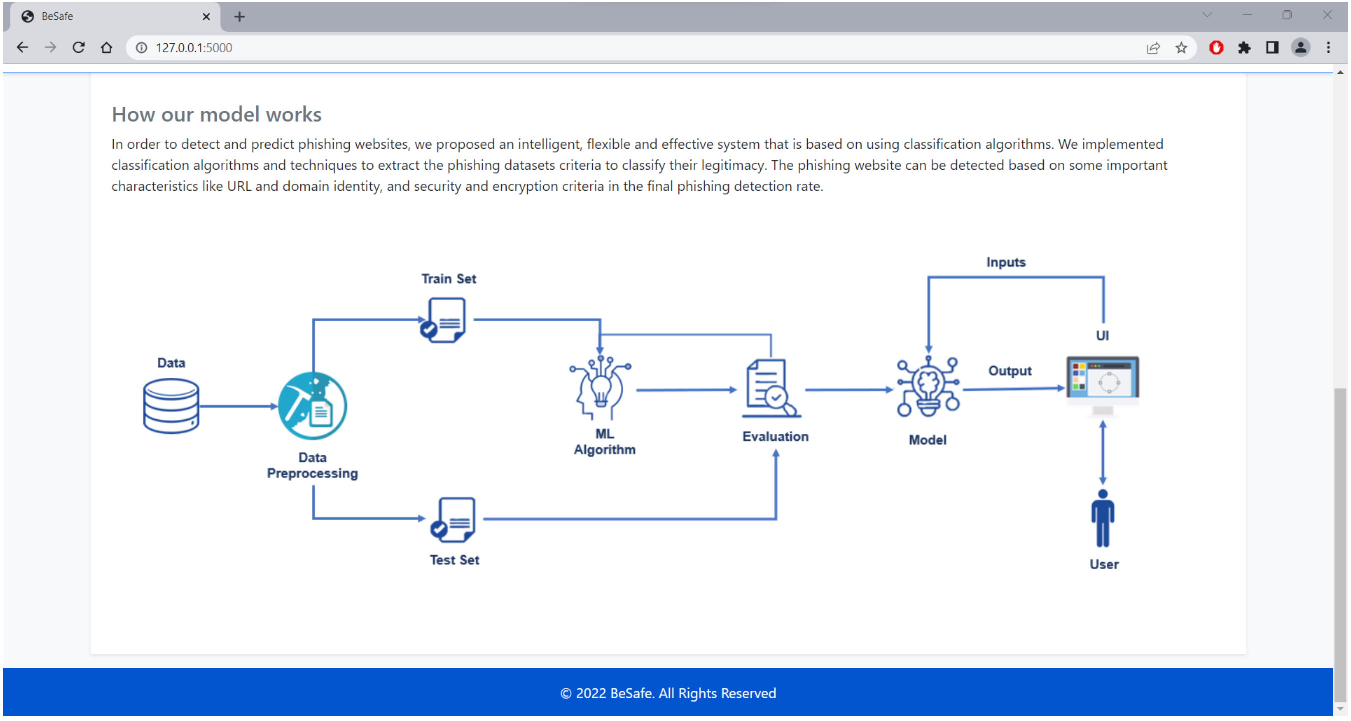
- Phishing is one of the most frequent forms of cyber crime, but despite how much we think we know about these scams, they still catch us out all too often.
- **The message is sent from a public email domain.** No legitimate organisation will send emails from an address that ends '@gmail.com'.
- **The domain name is misspelt.** There's another clue hidden in domain names that provides a strong indication of phishing scams – unfortunately, it complicates our previous clue.
- **The email is poorly written.** You can often tell if an email is a scam if it contains poor spelling and grammar.



Web Phishing Detection using Machine Learning

Our model has been trained using Logistic Regression with 91.6% Accuracy

© 2022 BeSafe. All Rights Reserved

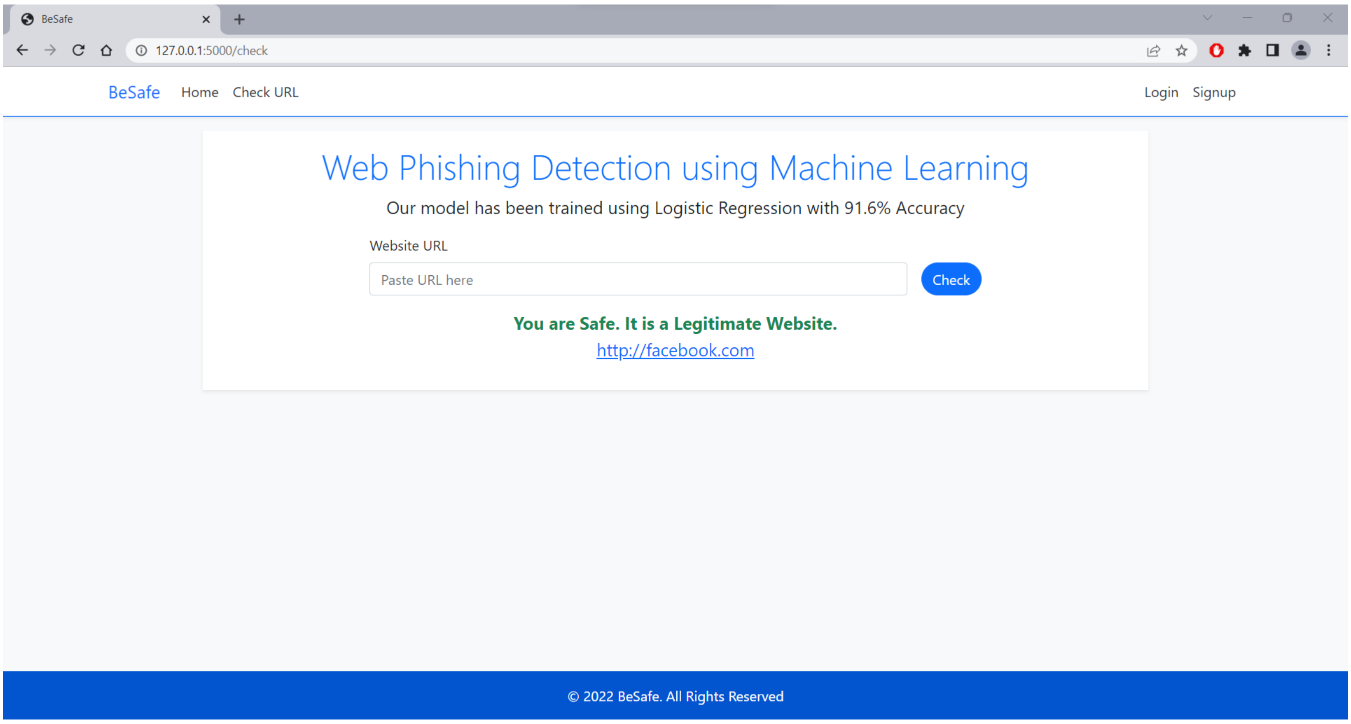


How our model works

In order to detect and predict 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 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.

© 2022 BeSafe. All Rights Reserved

Check URL Page



Web Phishing Detection using Machine Learning

Our model has been trained using Logistic Regression with 91.6% Accuracy

Website URL

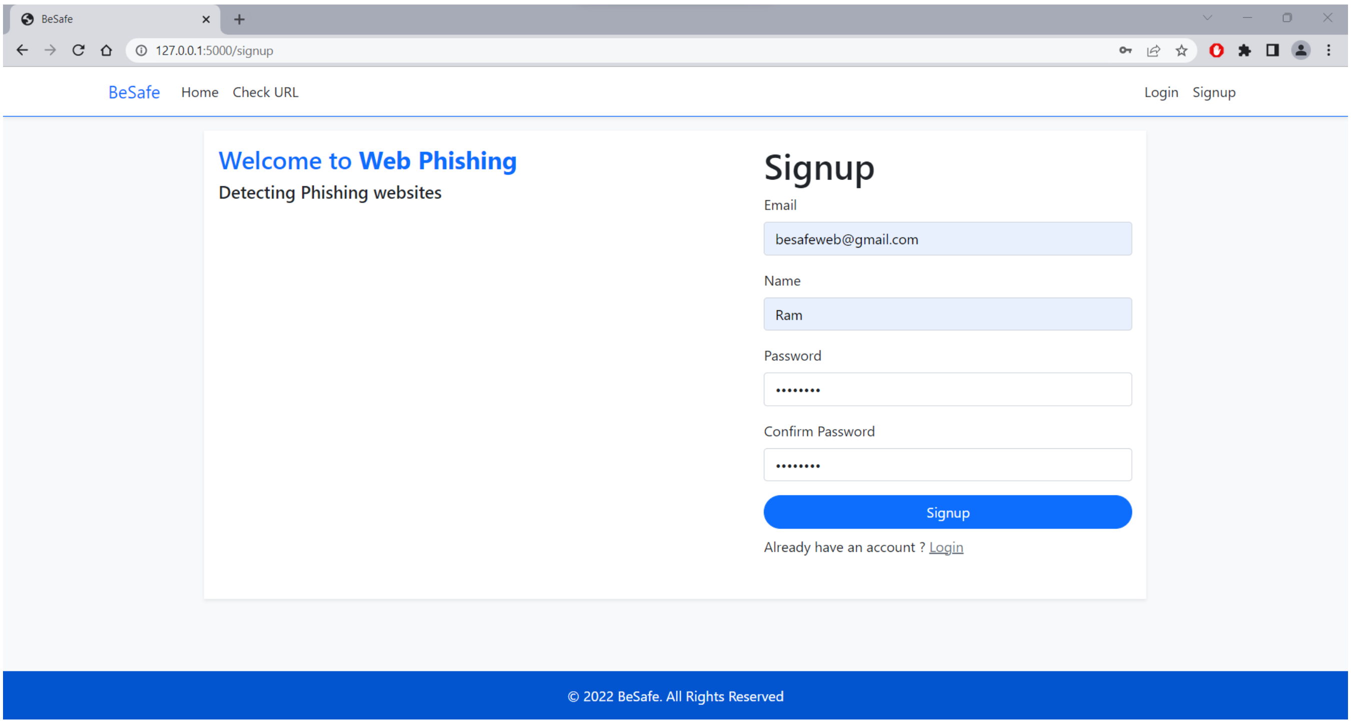
Paste URL here

You are Safe. It is a Legitimate Website.

http://facebook.com

© 2022 BeSafe. All Rights Reserved

Signup Page



Welcome to Web Phishing

Detecting Phishing websites

Signup

Email: besafeweb@gmail.com

Name: Ram

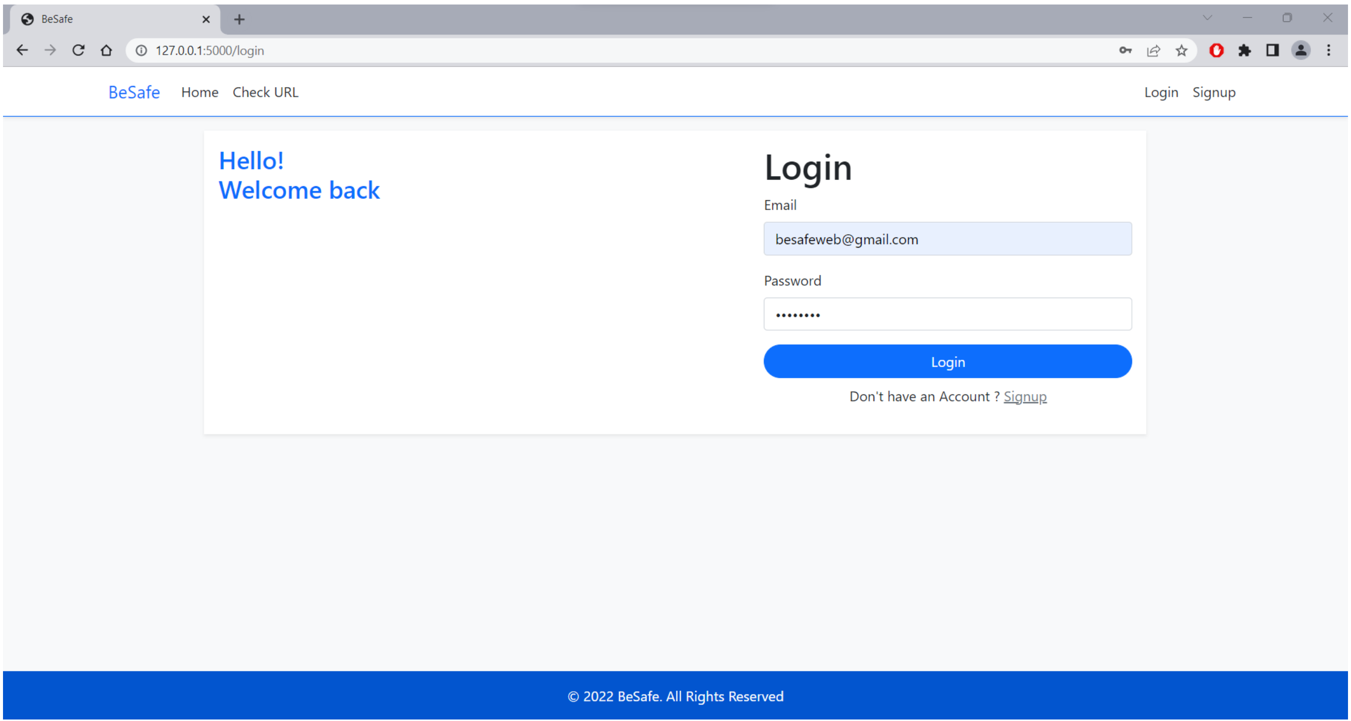
Password: ••••••••

Confirm Password: ••••••••

Already have an account ? Login

© 2022 BeSafe. All Rights Reserved

Email received



Hello!
Welcome back

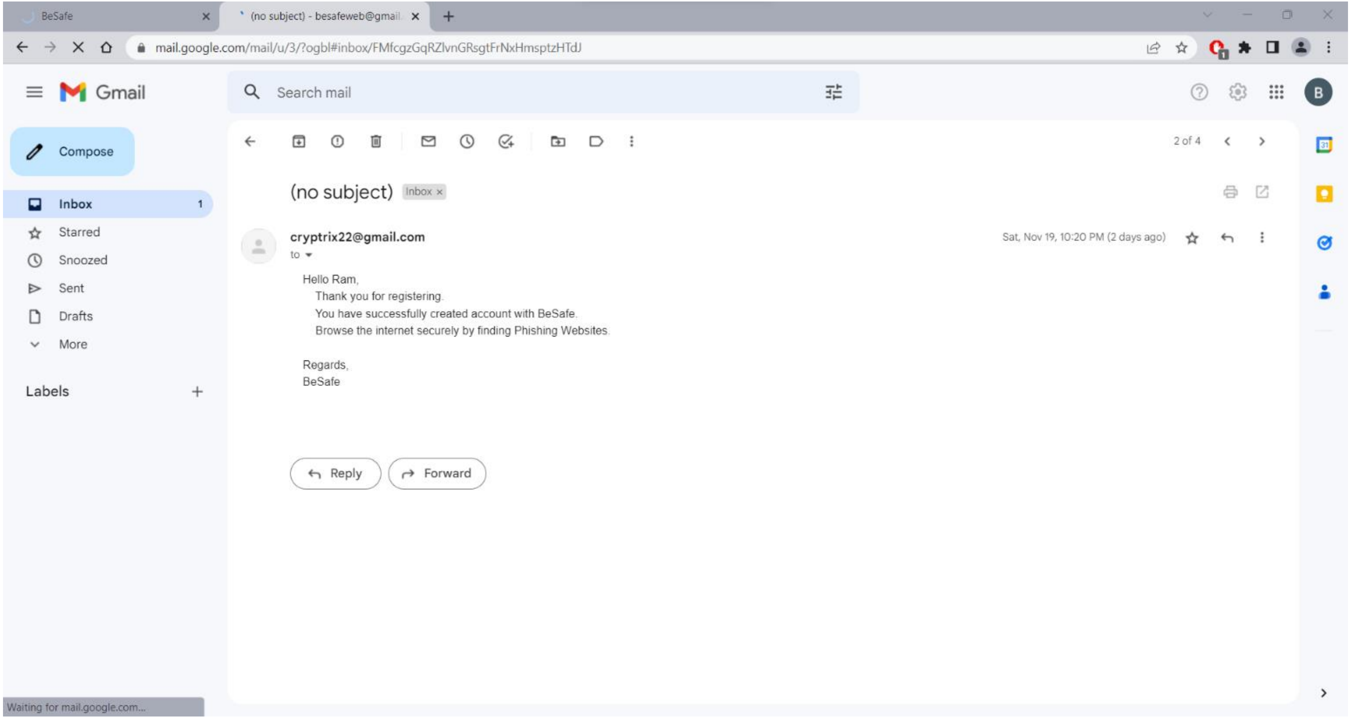
Login

Email: besafeweb@gmail.com

Password: ••••••••

Don't have an Account ? Signup

© 2022 BeSafe. All Rights Reserved


BeSafe
(no subject) - besafeweb@gmail.
mail.google.com/mail/u/3/?ogbl#inbox/FMfcgzGqRZlvnGRsgtFrNxHmsptzHTdJ
Gmail
Search mail
Compose
Inbox 1
Starred
Snoozed
Sent
Drafts
More
Labels
2 of 4
Inbox ×
Reply
Forward
Waiting for mail.google.com...

(no subject)

cryptrix22@gmail.com
to

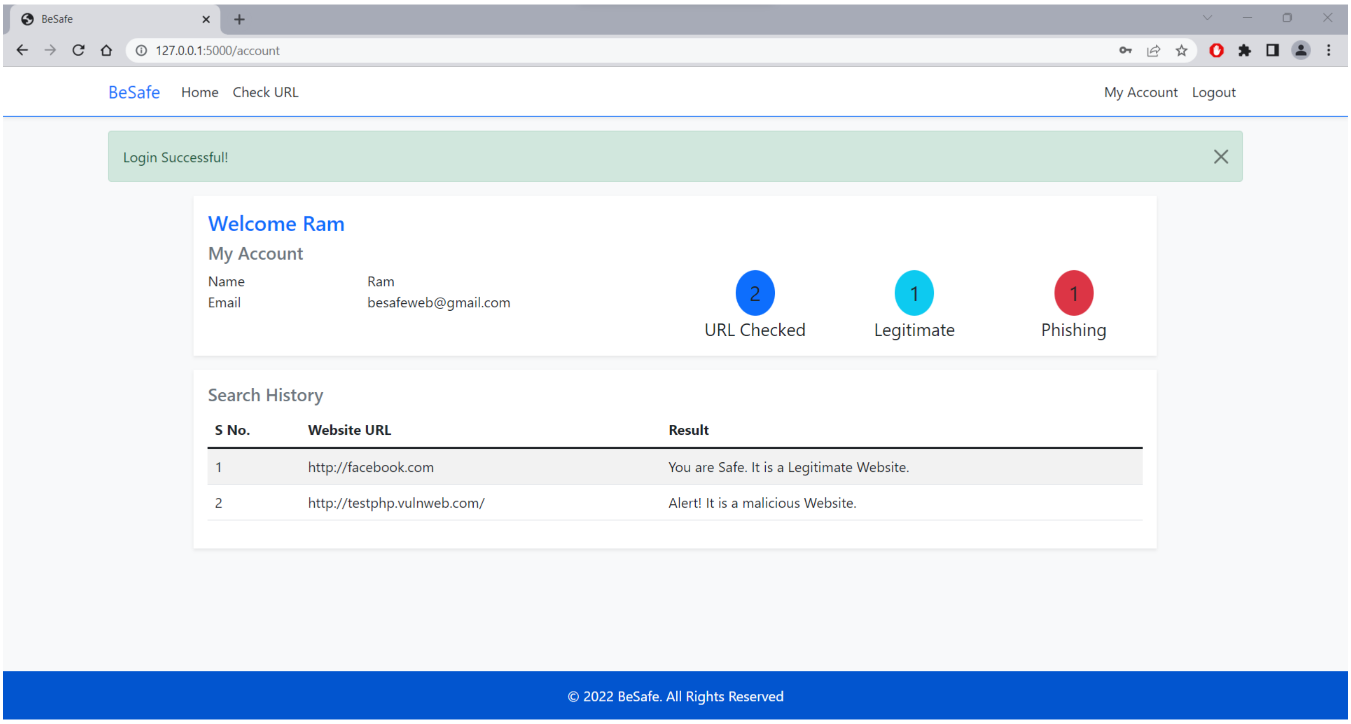
Sat, Nov 19, 10:20 PM (2 days ago)

Hello Ram,
Thank you for registering.
You have successfully created account with BeSafe.
Browse the internet securely by finding Phishing Websites.

Regards,
BeSafe

Login Page

My Account Page



Login Successful!

Welcome Ram

My Account

Name	Ram
Email	besafeweb@gmail.com

2 URL Checked

1 Legitimate

1 Phishing

Search History

S No.	Website URL	Result
1	http://facebook.com	You are Safe. It is a Legitimate Website.
2	http://testphp.vulnweb.com/	Alert! It is a malicious Website.

© 2022 BeSafe. All Rights Reserved