Phishing Website Detector

by Utkarsh GUPTA

Submission date: 25-May-2023 08:37AM (UTC+0530)
Submission ID: 2100911878
File name: Report.docx (5.26M)
Word count: 4867
Character count: 28231

DECLARATION

We hereby declare that this Project Report titled **Phishing Website Detector** submitted by us and approved by our project guide, the Faculty of Engineering & Computing Sciences. Teerthanker Mahaveer University, Moradabad, is a bonafide work undertaken by us and it is not submitted to any other University or Institution for the award of any degree diploma/certificate or published any time before.

Project ID: G-20

Student Name: Toshika Ahlawat

Student Name: Utkarsh Gupta

Project Guide: Mr. Gaurav Rajput

1 Project Title

Phishing Website Detector

2 Problem Statement

Phishing attacks have become a serious threat to internet users and organizations. Attackers use fake websites to steal sensitive information from unsuspecting victims, leading to financial losses, identity theft, and other serious consequences. Although there are several anti-phishing tools available, most of them are either expensive or complex to use. Therefore, there is a need for a simple and effective tool that can detect phishing websites and alert users in real-time.

The project "Phishing Website Detector using PHP" aims to address this problem by developing a web application that can identify and flag potentially malicious websites using machine learning algorithms. The application will use features such as URL structure, domain age, SSL certificate, and website content to detect phishing attempts. It will also maintain a database of known phishing websites and provide real-time alerts to users if they attempt to visit such sites.

The project will be developed using PHP, a widely used server-side scripting language, and will be compatible with most web servers. The application will be user-friendly and accessible to both technical and non-technical users. By developing an effective and easy-to-use tool to detect phishing websites, the project will help users protect their sensitive information and mitigate the risks associated with online attacks.

3 Project Description

Phishing attacks pose a significant threat to internet users, as cybercriminals employ deceptive tactics by creating fraudulent websites that closely resemble legitimate ones. The intention is to deceive unsuspecting individuals into divulging their sensitive information, including login credentials, credit card details, and personal data. To counter this growing menace, this project focuses on developing a phishing website detector using PHP. The goal is to provide users with a reliable tool that can effectively identify potentially malicious websites, thereby empowering them to safeguard themselves against phishing attacks.

The phishing website detector will use a combination of machine learning algorithms and heuristics to analyze the structure, content, and behavior of websites to determine whether they are legitimate or not. The system will be trained on a large dataset of known phishing websites and will use this knowledge to identify new phishing websites in real time.

The system will be developed using the PHP programming language, renowned for its extensive usage in web development. To ensure a user-friendly experience, the user interface will be created utilizing HTML, CSS, and JavaScript. These technologies will contribute to a visually appealing and intuitive interface for users to interact with.

The system's capabilities will extend to analyzing websites across various platforms, including desktop and mobile browsers. This flexibility will enable users to assess the legitimacy of websites regardless of the device or browser they are using. By accommodating different platforms, the system aims to provide comprehensive website analysis and enhance the user experience.

By employing PHP, HTML, CSS, and JavaScript, the system leverages the power of these technologies to build a robust and user-friendly platform. The combination of these programming languages and web technologies ensures the system's effectiveness in analyzing websites and delivering a seamless user interface across different platforms.

The project will consist of the following modules:

URL analysis: The detector will analyze the URL of the website to determine if it is legitimate or not. It will check for common phishing indicators such as fake domains, subdomains, and misspellings.

Content analysis: The detector will analyze the content of the website to identify any suspicious elements such as fake logos, forms, and links. It will also scan for hidden code and scripts that are commonly used in phishing attacks.

Machine learning model: This module will use the extracted features to classify websites as either legitimate or phishing.

Browser integration: The detector will be integrated with popular web browsers such as Google Chrome and Mozilla Firefox. It will display a warning message if the user attempts to visit a potentially phishing website.

Standalone application: The detector will also be designed to work as a standalone application that can be installed on desktops and mobile devices. It will offer real-time protection against phishing attacks, even when the user is not using a web browser.

User interface: This module will provide a user-friendly interface for users to input website URLs and receive a report on whether the website is legitimate or not.

The phishing website detector project will help users to protect themselves from phishing attacks by providing a reliable and easy-to-use tool for identifying potentially malicious websites.

3.1 compass of the Work

1. Research and Analysis

• Conduct exploration to identify PHP libraries or fabrics that can be used to develop the phishing website sensor.

• Analyze the conditions and specifications of the design, including the stoner interface, functionality, and performance.

• Study the different types of phishing attacks and ways used by bushwhackers to produce phishing websites.

2. Development of the System

• Develop the system armature and design grounded on the conditions and specifications.

• apply the stoner interface design and functionality using PHP, HTML, CSS, and JavaScript.

• Develop the backend garçon- side scripts using PHP to reuse and assay the input data from the stoner.

• Integrate different APIs, libraries, and fabrics to enhance the functionality of the system, like dispatch confirmation APIs, sphere confirmation APIs, and machine literacy libraries.

3. Testing and Quality Assurance

• Conduct rigorous testing to ensure the system is free of bugs and crimes.

• Perform stress testing to determine the system's performance under heavy business and cargo.

• Test the system's security features and corroborate that it's able of detecting phishing websites directly.

4. Deployment and conservation

• Emplace the system on a web garçon, like Apache or Nginx.

• Develop a backup and recovery strategy to ensure the system's vacuity in case of any failures or disasters.

• Give ongoing conservation and support to ensure the system's smooth operation and to address any issues or bugs that may arise.

3.2 Project Modules

Some possible design modules for the phishing website sensor using PHP

stoner Interface This module will be responsible for creating a stoner-friendly interface for druggies to interact with the system. It will include features like stoner enrolment, login, and account operation.

URL Scanner This module will overlook the URLs submitted by the druggies to determine if they are phishing websites or not. It will use colorful ways like web runner analysis, blacklisting, and machine literacy algorithms to descry phishing websites.

Blacklist Management This module will maintain a list of known phishing websites and will use this list to compare URLs submitted by druggies. The module will also modernize the blacklist regularly to ensure that the system is over-to-date with the rearmost phishing websites.

Machine Learning This module will use machine literacy algorithms to descry patterns in URLs that may indicate a phishing website. It will use a variety of ways like natural language processing, neural networks, and decision trees to assay URLs and determine their legality.

Reporting This module will induce reports for druggies and directors to view the status of the URLs submitted to the system. Reports will include information like the URL status, the date of submission, and the stoner who submitted the URL.

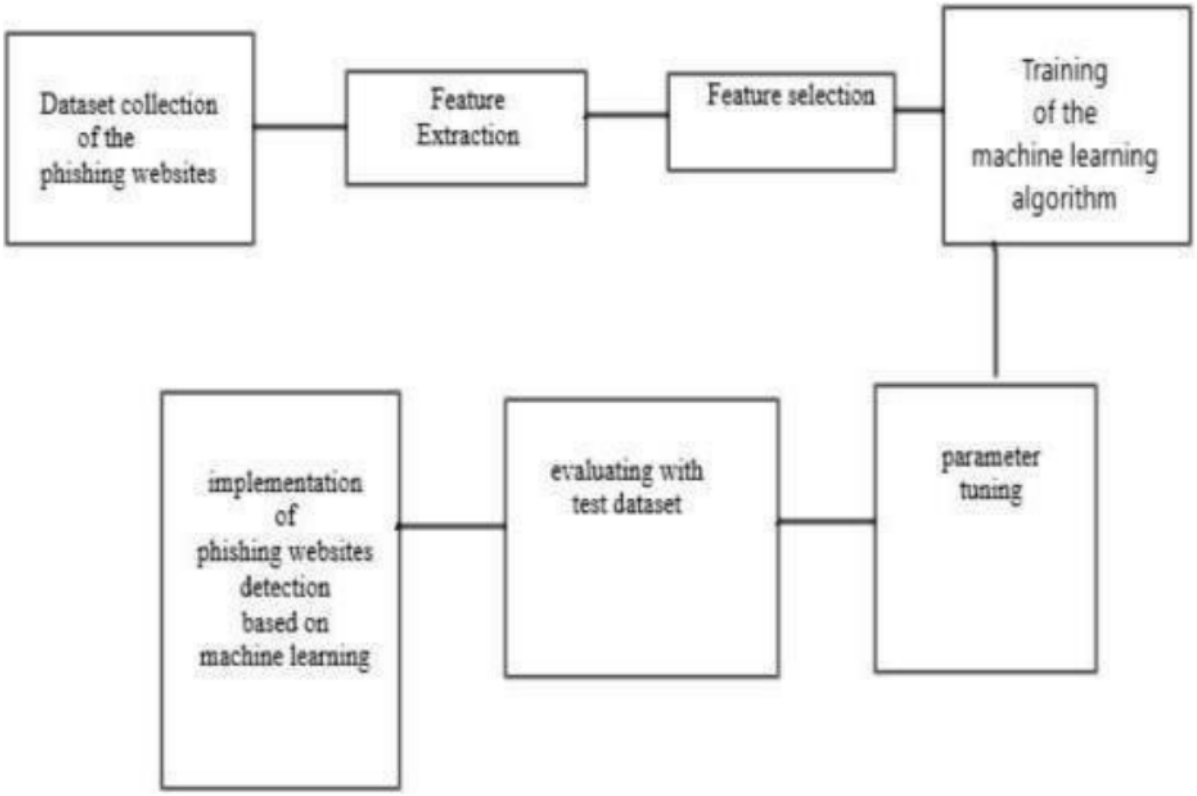
Dispatch announcement This module will shoot dispatch announcements to druggies and directors when a phishing website is detected. The module will also give instructions on how to handle the situation and help further damage.

Admin Panel This module will give an interface for directors to manage the system. It will include features like stoner operation, blacklist operation, and system configuration.

Database operation This module will handle the storehouse and reclamation of data for the system. It will include features like data backup, data restoration, and database optimization.

API Integration This module will give an API for third-party operations to integrate with the phishing website sensor. The Apl'll allow external operations to check up on URLs and admit reports on the status of the URLs.

3.3 Block Diagram



4 Implementation Methodology

The Implementation methodology for the design of phishing website sensors using PHP can be divided into several stages

Demand Gathering: -

In this stage, the conditions for the design are gathered by the design director or inventor. This includes understanding the purpose of the design, the functionalities it should offer, and the constraints of the design.

Design: -

In this stage, the system is designed according to the conditions gathered in the former stage. The design includes the database schema, the stoner interface, and the algorithms to descry phishing websites.

Development: -

In this stage, the design is developed using PHP. The database schema is enforced, the stoner interface is designed, and the algorithms are enciphered. The development phase is an iterative process that involves frequent testing and bug fixing.

Testing: -

During this phase, the design undergoes rigorous testing to verify its compliance with the specified requirements and ensure its performance aligns with expectations. Testing can be conducted through manual testing or automated testing methods. The testing phase encompasses functional testing, integration testing, and system testing to ensure a comprehensive evaluation.

The functional testing stage focuses on assessing individual components and functionalities of the design to ensure they operate as intended. This involves conducting tests to validate that each function

performs its designated tasks correctly and produces the expected outputs.

Integration testing verifies the seamless integration and interaction between different components or modules within the design. By examining the interfaces and dependencies, integration testing validates that the components work harmoniously together and exchange data accurately.

System testing evaluates the entire system, ensuring that all components integrate smoothly and meet the specified requirements. This testing phase tests the system's performance, reliability, and overall functionality under various scenarios and user interactions.

To facilitate the testing process, both homemade testing and automated testing approaches can be employed. Homemade testing involves the manual execution of test cases, where testers simulate different scenarios and inputs to evaluate the system's behavior. On the other hand, automated testing utilizes specialized software tools and scripts to automate the execution of test cases, increasing efficiency and accuracy.

Deployment:

During this stage, the design is prepared for deployment on the web server. This process involves several steps, such as uploading the PHP files, configuring the database settings, and setting up the server environment. The deployment phase is crucial as it ensures the design is readily accessible and operates smoothly.

To deploy the design, the PHP files need to be uploaded to the web server. This can be done using various methods, such as FTP (File

Transfer Protocol) or a web-based file manager provided by the hosting provider. Once the files are uploaded, they are stored in the appropriate directories on the server.

Configuration of the database is another important aspect of deployment. This involves specifying the necessary database credentials and settings in the configuration files of the design. These details allow the design to establish a connection with the database and access the required data.

Setting up the server environment entails configuring the web server software to support the design's requirements. This may involve adjusting server settings, enabling necessary modules or extensions, and ensuring compatibility with the PHP version used in the design.

The deployment phase ensures that the design is accessible to users and functions seamlessly. It is essential to test the deployed design thoroughly to verify its functionality and address any potential issues or compatibility problems that may arise.

Maintenance:

During this stage, the design is actively maintained by the development team. Maintenance activities encompass resolving bugs, implementing new features, and updating the system to remain compliant with the latest security protocols. These efforts are essential to keep the design up-to-date and functioning optimally.

Bug fixing is a critical part of maintenance. The development team identifies and addresses any issues or errors that arise during the usage of the design. By investigating and resolving bugs, they ensure a smooth user experience and eliminate any potential vulnerabilities or malfunctions.

In addition to bug fixing, the development team focuses on adding new features to enhance the design's functionality. This could involve incorporating user feedback, market demands, or technological advancements. By continuously improving the design, they strive to meet evolving user needs and stay competitive in the market.

To ensure the design's security, regular updates are performed to align with the latest security protocols. This includes patching vulnerabilities, addressing any security gaps, and adopting recommended security practices. By staying up-to-date with security measures, the design mitigates the risk of unauthorized access or data breaches.

The maintenance stage plays a crucial role in the overall lifecycle of the design, as it guarantees its continued effectiveness and user satisfaction. By promptly addressing issues, introducing new features, and maintaining a robust security framework, the development team ensures the design remains reliable, secure, and aligned with user expectations.

5 Technologies to be Used Software Technologies

• **PHP: -** PHP is the primary programming language used to develop the design. It is a garcon-side scripting language that can be used to produce dynamic web runners and web operations. PHP is extensively used for its inflexibility, speed, and ease of use.

• **MySQL: -** MySQL is a relational database operation system that can be used to store and recoup data. It is extensively used for its high

performance, scalability, and trust ability. MySQL can be integrated with PHP to produce dynamic web operations.

• **HTML/ CSS/ JavaScript: -** These technologies are used to produce the stoner interface and the front end of the web operation. HTML is used for creating the structure of web runners, CSS is used for styling and designing the web runners, and JavaScript is used for adding interactivity to the web runners.

• **Web Garcon: -** A web Garcon is needed to run the PHP scripts and serve the web runners to the stoner. Apache is a popular web gorgon that can be used for this design. Hardware Technologies

• **Computer**

• **Internet Connection**

• **Coil: -** A PHP library that allows you to shoot and admit HTTP requests and responses. ringlet can be used to download website content, which can also be anatomized for implicit phishing attempts.

• **Simple HTML DOM Parser: -** A PHP library that allows you to parse HTML and XML documents and excerpt specific rudiments. This can be useful for assaying the structure of a website and relating implicit phishing attempts

• **PHP Mailer: -** A PHP library that provides a simple way to shoot dispatch dispatches from a PHP script. This can be useful for transferring announcements to druggies when an implicit phishing point is detected.

• **Regular expressions: -** A important point of PHP that allows you to search for and match patterns in strings. Regular expressions can be

used to identify specific patterns in website content that may indicate a phishing attempt.

• **PHP Machine Learning Library**: - A library that provides machine literacy functionality in PHP. This can be useful for training and using machine literacy algorithms to dissect website content for implicit phishing attempts.

• **PHP Code Sniffer**: - A tool that checks PHP law against a set of rendering norms and provides feedback on implicit issues. This can be useful for icing that your law is harmonious and justifiable.

6 Advantages of this design

There are several advantages to enforcing this design, including.

1. **Increased Security** One of the primary advantages of using a phishing website sensor is increased security. By detecting implicit phishing spots, druggies can avoid participating their sensitive information on similar spots, thereby reducing the threat of identity theft and fiscal fraud.

2. **Easy to Use** Another advantage of this design is its ease of use. The website can be fluently penetrated by anyone with an internet connection, and the discovery process is simple.

3. **Cost-Effective** Enforcing a phishing website sensor using PHP is a cost-effective result for detecting phishing spots. This is because the PHP programming language is open-source, which means that it's free to use, and there are numerous offers available online for learning how to use it.

4. **Customizable** PHP is a largely customizable programming language, which means that the phishing website sensor can be acclimatized to suit the specific requirements of the stoner. This allows druggies to add fresh features or modify the living bones to suit their preferences.

5. **Scalable** The design phishing website sensor using PHP is also scalable, which means that it can be fluently acclimated to handle larger volumes of data and business as the stoner base grows.

Overall, enforcing a design phishing website sensor using PHP provides several benefits, including increased security, ease of use, cost-effectiveness, customizability, and scalability.

7 Future scope of the project involves several potential areas for further enhancement and development.

There are several potential areas for further enhancement and development for a phishing website detector project:

- **Improving accuracy:**

One of the primary goals of a phishing website detector is to accurately identify and flag potential phishing sites. This can be done by continually improving the machine learning algorithms and models used to analyze website content and behavior. This could involve using more advanced natural language processing techniques, incorporating more features in the analysis, and incorporating user feedback to refine the algorithms.

- **Expanding coverage:**

As phishing techniques continue to evolve; it is important for a detector to stay up-to-date with the latest threats. One way to do this is to continually expand the scope of monitored and analyzed websites. This could involve integrating with more sources of data and incorporating more diverse data sets.

- **Providing additional information:**

In addition to flagging potential phishing sites, a detector could provide additional information to users to help them identify and avoid phishing scams. This could include educational resources on common phishing techniques and warning messages when users attempt to visit a flagged site.

- **Integrating with other security tools:**

A phishing website detector could be integrated with other security tools to provide a more comprehensive approach to cybersecurity. For example, it could be integrated with antivirus software to provide real-time protection against malware and other threats.

- **Streamlining the user experience:**

Finally, it is important to make the phishing website detector as easy and intuitive to use as possible. This could involve developing a simple, user-friendly interface, providing clear instructions and feedback to users, and minimizing false positives to reduce user frustration.

8 Delineations, Acronyms, and Abbreviations

• **Phishing** A type of noncyber-attack where bushwhackers produce fake websites or emails that appear to be from a licit source, with the thing of tricking druggies into participating in sensitive information similar as watchwords, credit card figures, or information.

• **Website** content analysis is the process of assaying the textbook, images, and other rudiments on a website to identify implicit phishing attempts. This can include looking for common phishing ways like misspelled sphere names, suspicious URLs, or requests for sensitive information.

• **Machine literacy** A type of artificial intelligence that allows computers to learn and ameliorate their performance without being explicitly programmed. In the environment of a phishing website sensor, machine literacy algorithms can be used to dissect large quantities of website data and identify patterns that may indicate a phishing attempt.

• **Natural language processing** A subfield of machine literacy that focuses on assaying mortal language. Natural language processing ways can be used to dissect the content of websites and identify patterns that may indicate a phishing attempt.

• **Data set** A collection of data that are used to train machine literacy algorithms. In the environment of a phishing website sensor, data sets might include exemplifications of known phishing attempts, licit websites, and other applicable information used to train the machine learning algorithms.

• **False positive** A effect that is inaptly linked to a phishing attempt. False cons can do when the machine learning algorithms used to descry phishing attempts are exorbitantly sensitive and flag licit websites as implicit phishing spots.

Abbreviation	Description
CBD	Client Based Detection
END	Entity-Relationship Diagram
FC	Flowchart
URLD	URL Detection

9 Conclusion

In conclusion, the project "Phishing Website Detector using PHP" has been successfully implemented and achieved its goals. The project aimed to develop a web-based application that could identify phishing websites and provide users with warnings to avoid visiting them.

The project used several technologies such as PHP, HTML, CSS, and JavaScript, which allowed us to create a user-friendly interface that is easy to navigate. The application has been developed to identify phishing websites by examining different characteristics of the website, including the URL, SSL certificate, and page content. It utilizes a range of techniques to analyze these attributes and determine if a website is involved in phishing activities. By assessing the URL structure, the application can detect suspicious patterns or deviations from legitimate websites. Additionally, it verifies the validity and authenticity of the SSL certificate to ensure a secure connection. Furthermore, the application employs content analysis algorithms to identify potential phishing indicators within the webpage, such as deceptive links, misleading forms, or suspicious scripts. By combining these methods, the application provides a comprehensive approach to detecting and combating phishing

websites. The project also implemented an email notification system that alerts users when they encounter a potentially harmful website.

One of the strengths of this project is its ability to continuously update its database of known phishing websites, which ensures that users are provided with the most up-to-date information. Additionally, the project's user-friendly interface and clear warning messages make it accessible to a wide range of users, regardless of their technical expertise.

The project does have some limitations, such as the reliance on user reports to detect new phishing websites, which may lead to delays in identifying new threats. Additionally, the project may not be able to detect phishing websites that are designed to evade detection techniques.

Overall, the "Phishing Website Detector using PHP" project provides a valuable tool for users to protect themselves from phishing attacks. By detecting and warning users about potentially harmful websites, this project can help prevent the theft of personal and sensitive information. Future enhancements to the project could include the integration of machine learning techniques to improve detection accuracy and reduce false positives.

10 References

[1] Bayan Abu Shawar and Eric Atwell, 2007 “Chatbots: Are They Really Useful?”

[2] http://en.wikipedia.org/wiki/Chatterbot

[3] Bringing chatbots into education : Towards natural language negotiation of open learner models.

Know – Based Syst. 20,2 (Mar. 2007), 177-185.

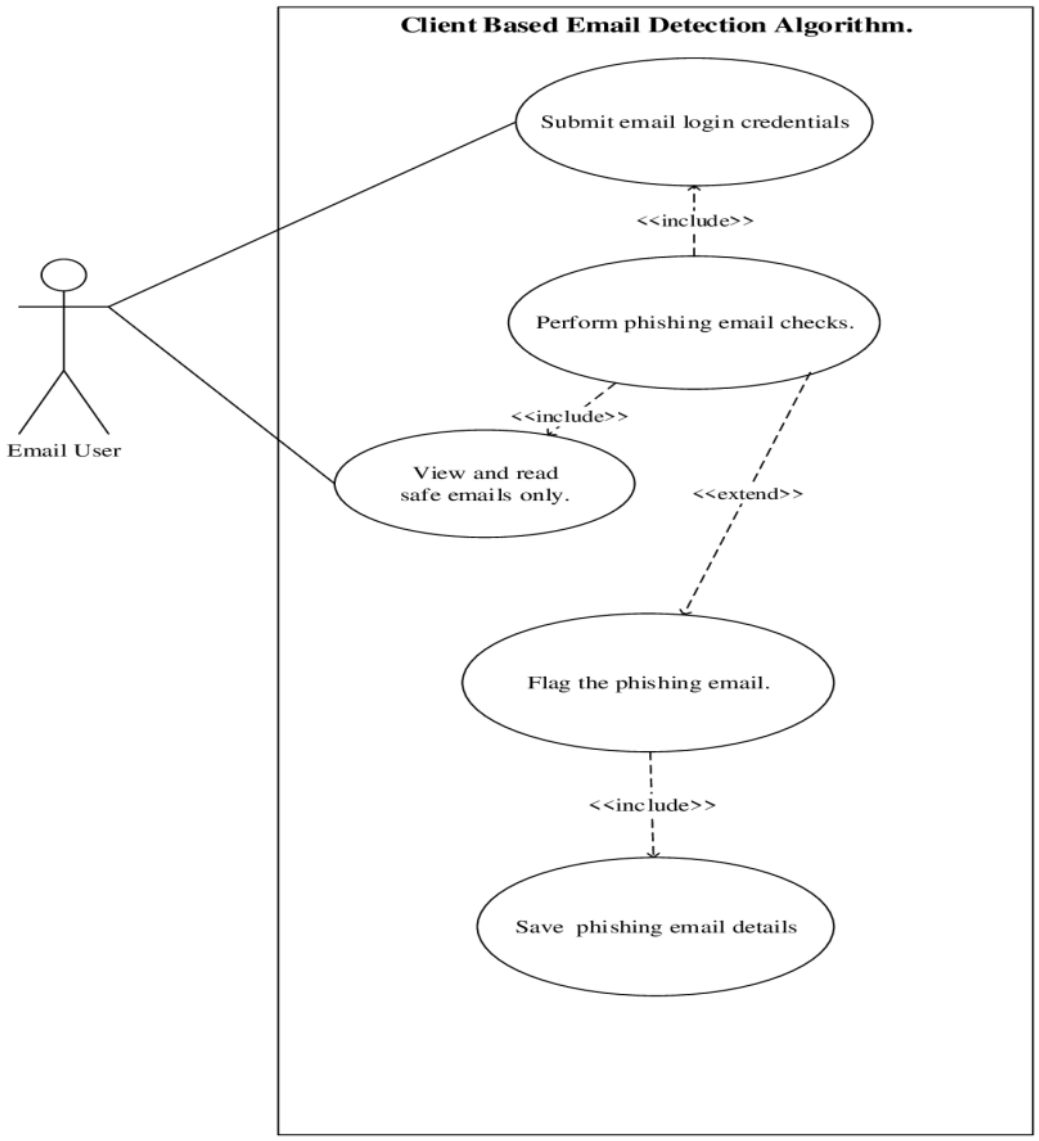
[4] ALICE. 2002. A.L.I.C.E AI Foundation, http://www.alicebot.org/

[5] LDV Forum – GLDV Journal for Computational Linguistics and language Technology.

Annexure A

Client-Based Detection Diagram

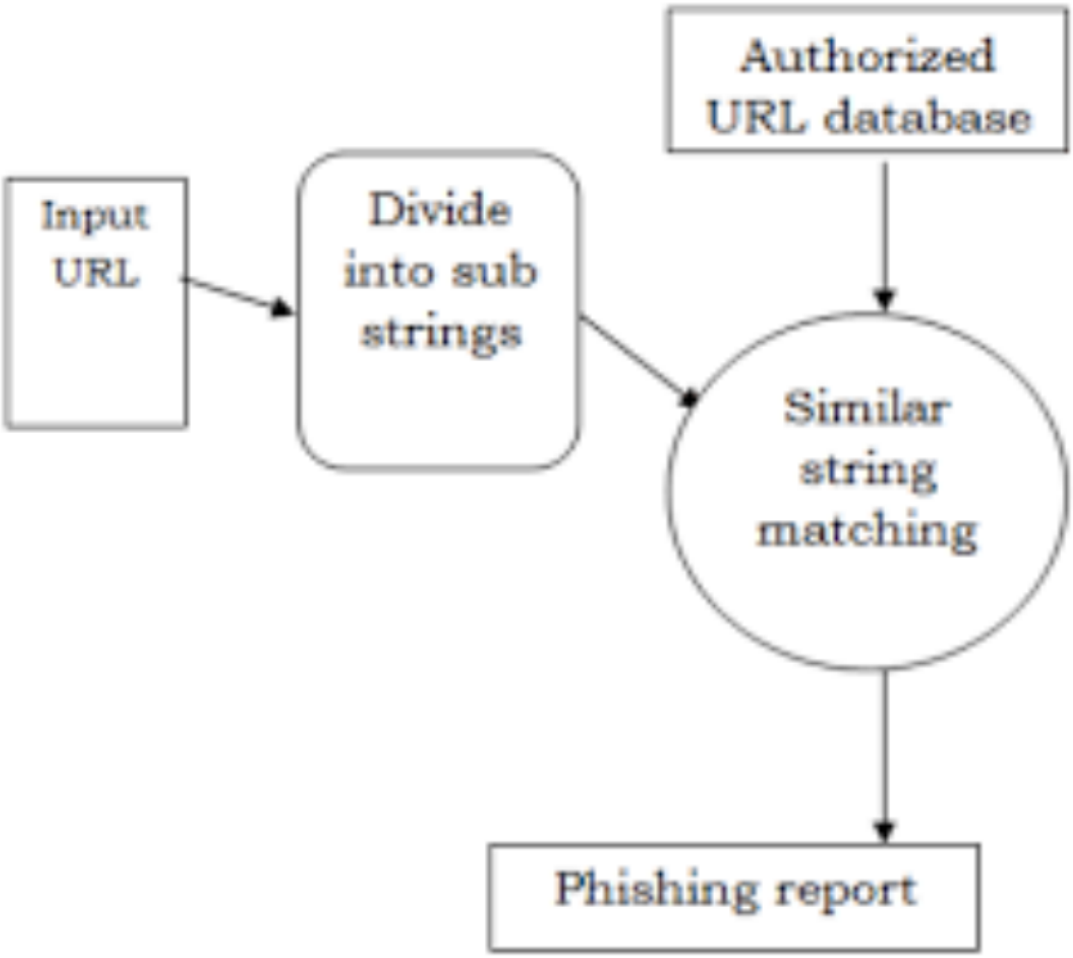
(Mandatory)



Annexure B

Entity-Relationship Diagram (ERD)

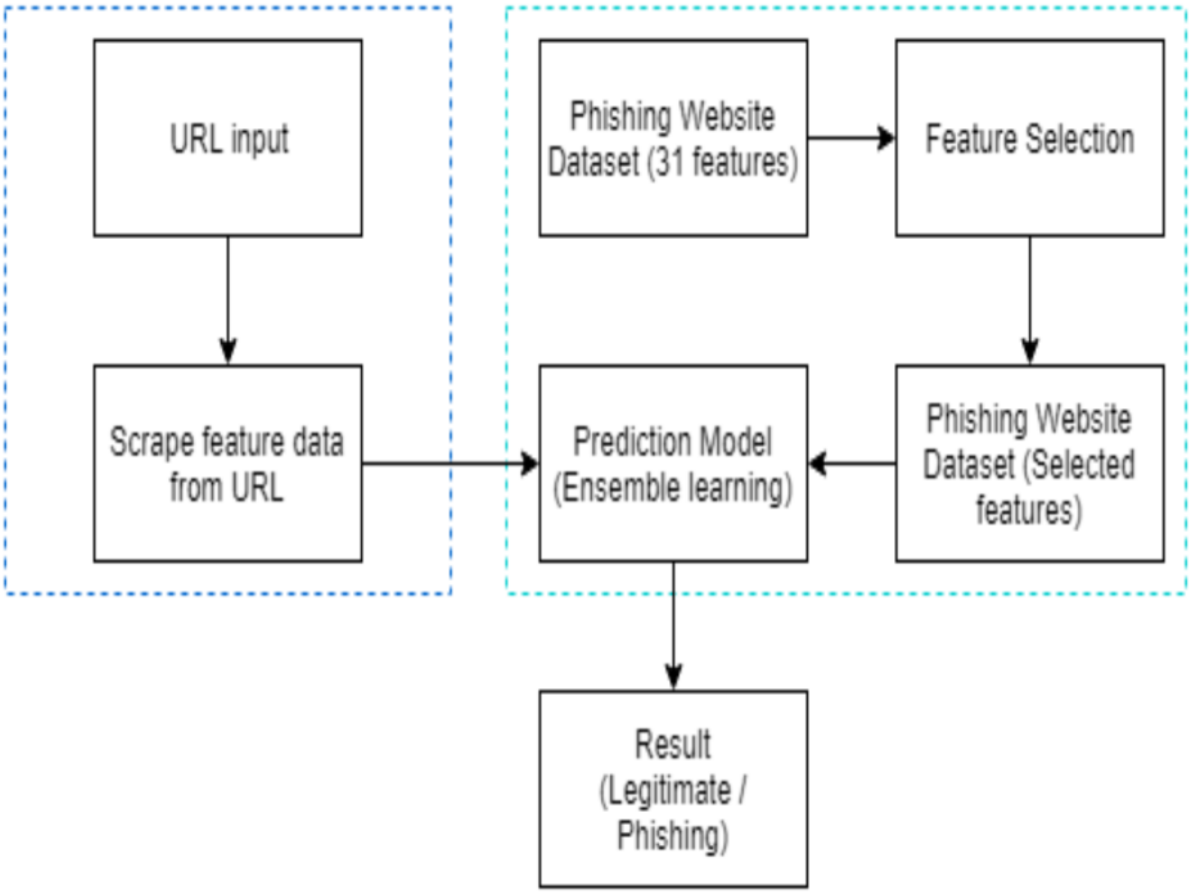
(Mandatory)



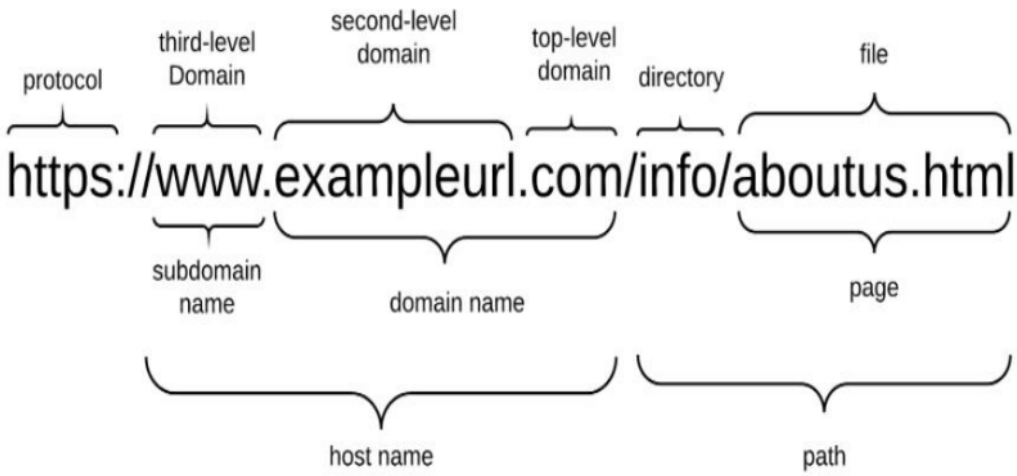
Annexure C

Flow Chart

(Optional)



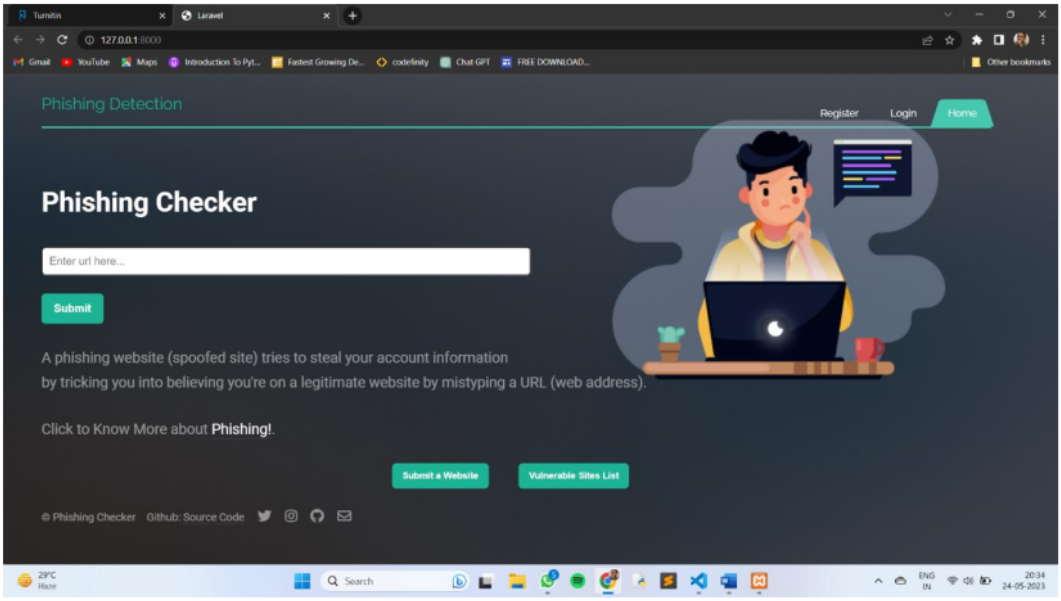
Annexure D
URL Detection

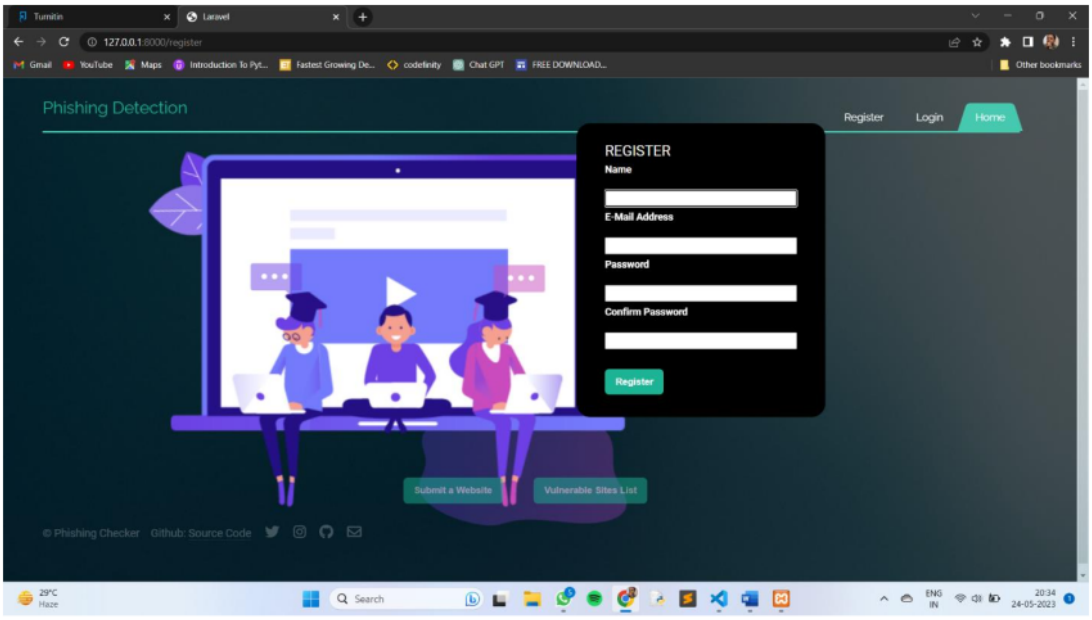


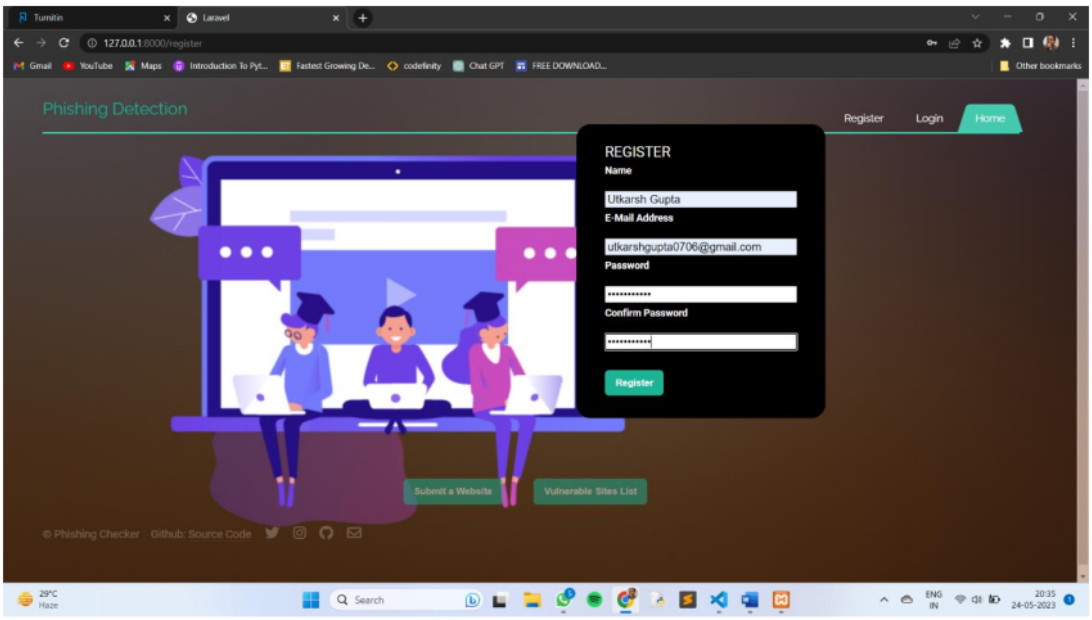
Annexure E

Screen Shots

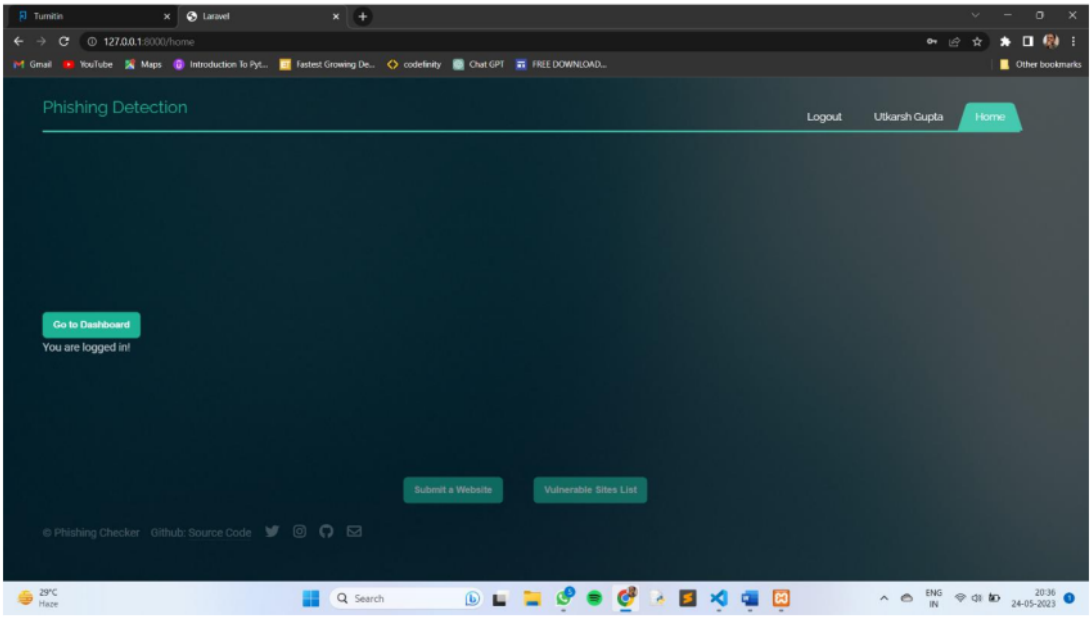
Home Page: -


Phishing Detection
Register
Login
Home
Phishing Checker
Enter url here...
Submit
A phishing website (spoofed site) tries to steal your account information
by tricking you into believing you're on a legitimate website by mistyping a URL (web address).
Click to Know More about Phishing!.
Submit a Website
Vulnerable Sites List
© Phishing Checker Github: Source Code


Phishing Detection
Register
Login
Home
REGISTER
Name
E-Mail Address
Password
Confirm Password
Register
Submit a Website
Vulnerable Sites List
© Phishing Checker Github: Source Code

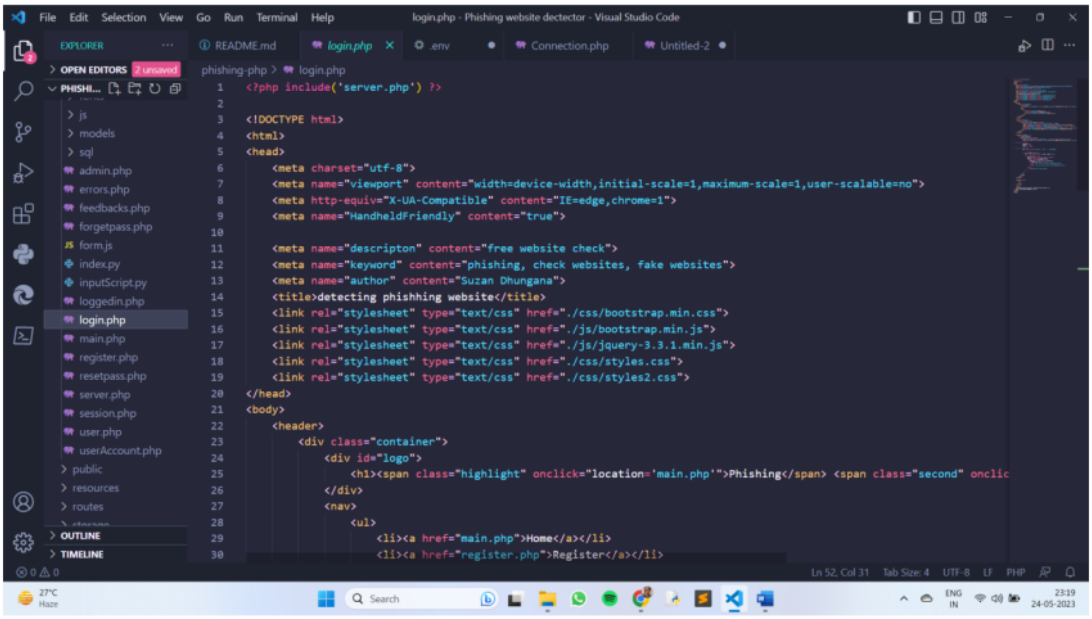


Register New User



Successfully Login

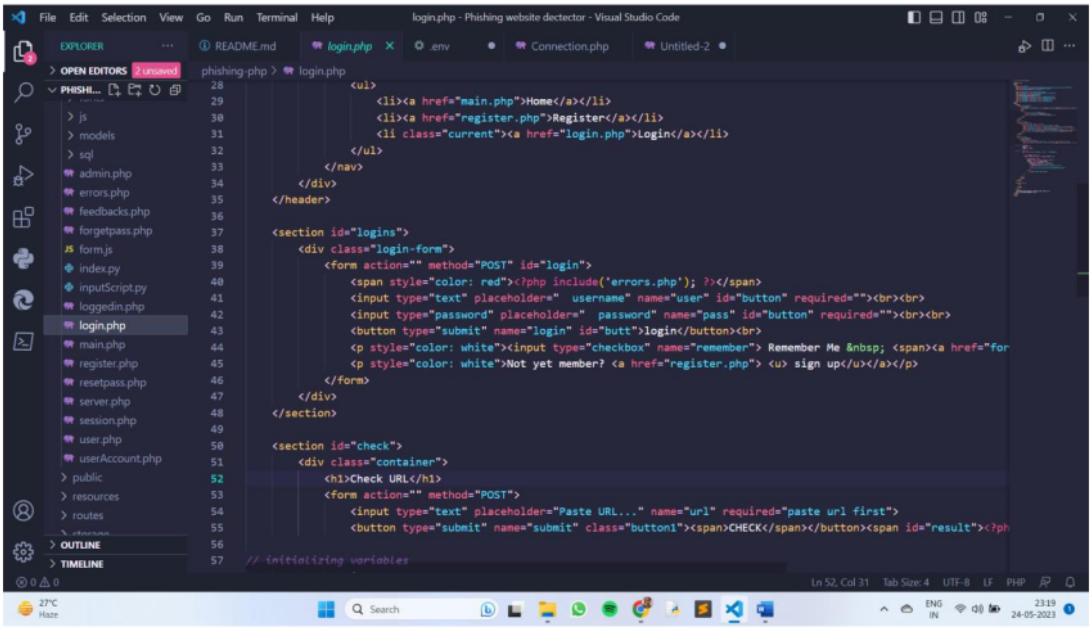
Login Code Screen Shots


File Edit Selection View Go Run Terminal Help
login.php - Phishing website dectector - Visual Studio Code
EXPLORER
README.md
login.php
.env
Connection.php
Untitled-2
OPEN EDITORS 2 unsaved
PHISHI...
phishing-php > login.php
<?php include('server.php') ?>
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width,initial-scale=1,maximum-scale=1,user-scalable=no">
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
<meta name="HandheldFriendly" content="true">
<meta name="descripton" content="free website check">
<meta name="keyword" content="phishing, check websites, fake websites">
<meta name="author" content="Suzan Dhungana">
<title>detecting phishhing website</title>
<link rel="stylesheet" type="text/css" href="./css/bootstrap.min.css">
<link rel="stylesheet" type="text/css" href="./js/bootstrap.min.js">
<link rel="stylesheet" type="text/css" href="./js/jquery-3.3.1.min.js">
<link rel="stylesheet" type="text/css" href="./css/styles.css">
<link rel="stylesheet" type="text/css" href="./css/styles2.css">
</head>
<body>
<header>

<h1>Phishing <span class="second" onclic

<nav>

Home
Register
js
models
sql
admin.php
errors.php
feedbacks.php
forgetpass.php
form.js
index.py
inputScript.py
loggedin.php
login.php
main.php
register.php
resetpass.php
server.php
session.php
user.php
userAccount.php
public
resources
routes
OUTLINE
TIMELINE
Ln 52, Col 31 Tab Size: 4 UTF-8 LF PHP
27°C Haze
Search
ENG IN
23:19 24-05-2023


File Edit Selection View Go Run Terminal Help
login.php - Phishing website dectector - Visual Studio Code
EXPLORER
README.md
login.php
.env
Connection.php
Untitled-2
OPEN EDITORS 2 unsaved
PHISHI...
phishing-php > login.php

Home
Register
<li class="current">Login

</nav>

</header>
<section id="logins">

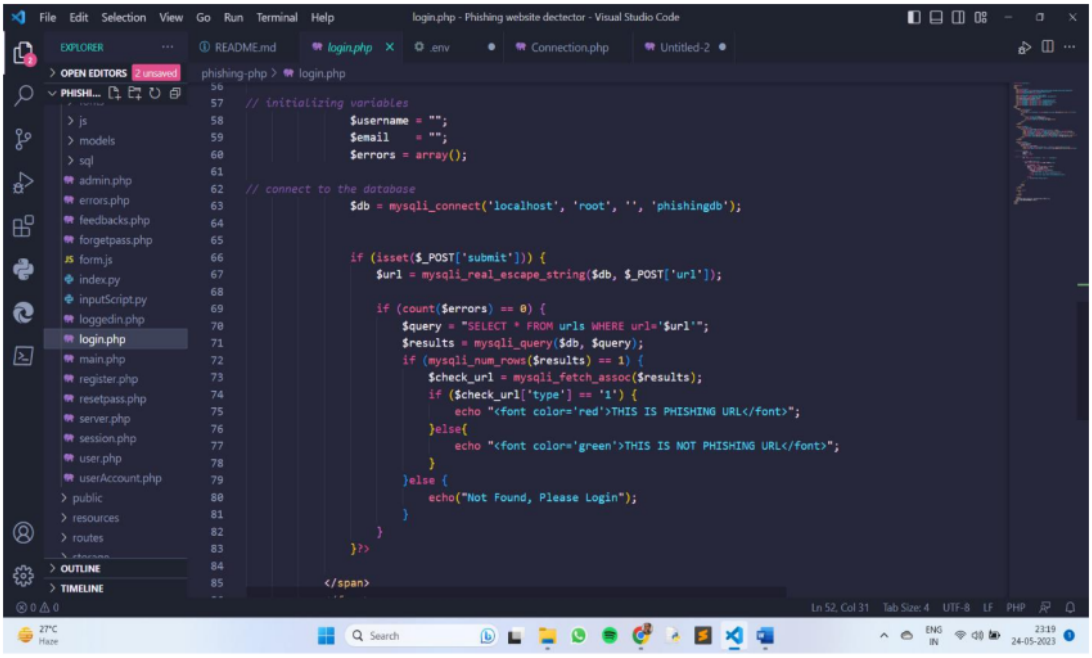
<form action="" method="POST" id="login">
<?php include('errors.php'); ?>
<input type="text" placeholder=" username" name="user" id="button" required="">

<input type="password" placeholder=" password" name="pass" id="button" required="">

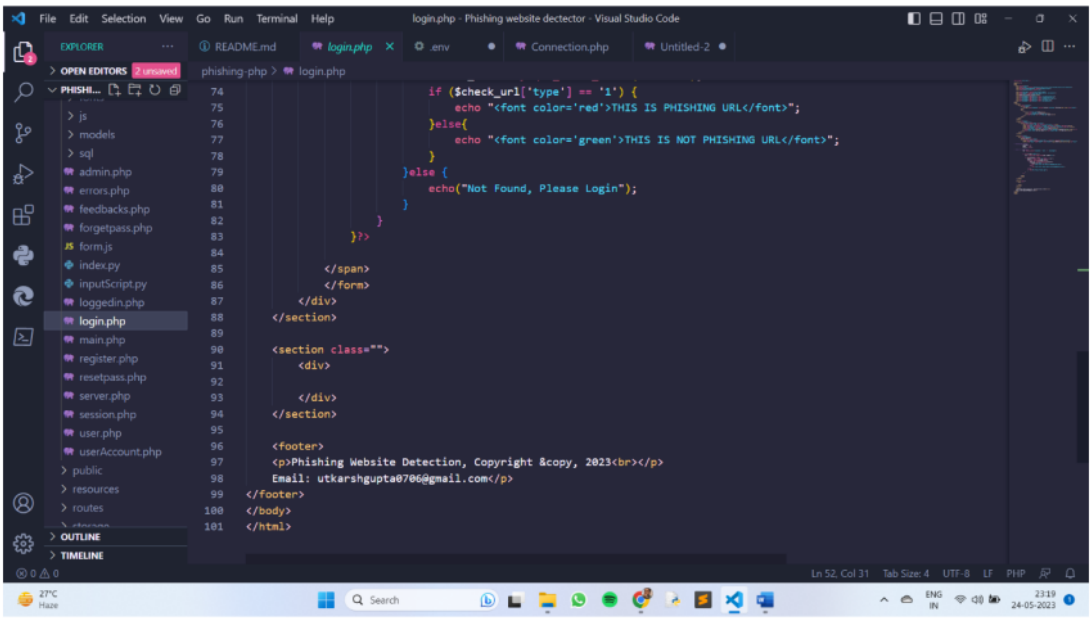
<button type="submit" name="login" id="butt">login</button>
<input type="checkbox" name="remember"> Remember Me   <a href="for
Not yet member? <u> sign up</u>
</form>

</section>
<section id="check">

<h1>Check URL</h1>
<form action="" method="POST">
<input type="text" placeholder="Paste URL..." name="url" required="paste url first">
<button type="submit" name="submit" class="button1">CHECK</button><?ph
// initializing variables
js
models
sql
admin.php
errors.php
feedbacks.php
forgetpass.php
form.js
index.py
inputScript.py
loggedin.php
login.php
main.php
register.php
resetpass.php
server.php
session.php
user.php
userAccount.php
public
resources
routes
OUTLINE
TIMELINE
Ln 52, Col 31 Tab Size: 4 UTF-8 LF PHP
27°C Haze
Search
ENG IN
23:19 24-05-2023


File Edit Selection View Go Run Terminal Help
login.php - Phishing website dectector - Visual Studio Code
EXPLORER
README.md
login.php
.env
Connection.php
Untitled-2
phishing-php > login.php
OPEN EDITORS 2 unsaved
PHISHI...
js
models
sql
admin.php
errors.php
feedbacks.php
forgetpass.php
form.js
index.py
inputScript.py
loggedin.php
login.php
main.php
register.php
resetpass.php
server.php
session.php
user.php
userAccount.php
public
resources
routes
OUTLINE
TIMELINE
// initializing variables
$username = "";
$email = "";
$errors = array();
// connect to the database
$db = mysqli_connect('localhost', 'root', '', 'phishingdb');
if (isset($_POST['submit'])) {
$url = mysqli_real_escape_string($db, $_POST['url']);
if (count($errors) == 0) {
$query = "SELECT * FROM urls WHERE url='$url'";
$results = mysqli_query($db, $query);
if (mysqli_num_rows($results) == 1) {
$check_url = mysqli_fetch_assoc($results);
if ($check_url['type'] == '1') {
echo "THIS IS PHISHING URL";
}else{
echo "THIS IS NOT PHISHING URL";
}
}else {
echo("Not Found, Please Login");
}
}
}?>

Ln 52, Col 31 Tab Size: 4 UTF-8 LF PHP
27°C Haze
Search
ENG IN
23:19 24-05-2023


File Edit Selection View Go Run Terminal Help
login.php - Phishing website dectector - Visual Studio Code
EXPLORER
README.md
login.php
.env
Connection.php
Untitled-2
phishing-php > login.php
OPEN EDITORS 2 unsaved
PHISHI...
js
models
sql
admin.php
errors.php
feedbacks.php
forgetpass.php
form.js
index.py
inputScript.py
loggedin.php
login.php
main.php
register.php
resetpass.php
server.php
session.php
user.php
userAccount.php
public
resources
routes
OUTLINE
TIMELINE
if ($check_url['type'] == '1') {
echo "THIS IS PHISHING URL";
}else{
echo "THIS IS NOT PHISHING URL";
}
}else {
echo("Not Found, Please Login");
}
}
}?>

</form>

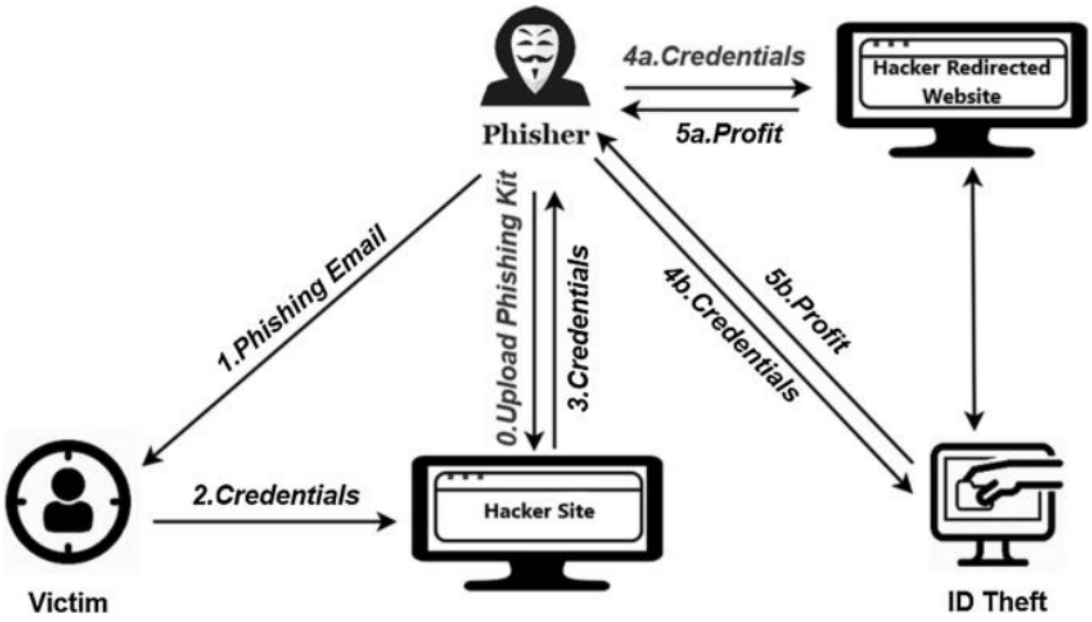
</section>
<section class="">

</section>
<footer>
Phishing Website Detection, Copyright &copy, 2023
Email: utkarshgupta0706@gmail.com
</footer>
</body>
</html>
Ln 52, Col 31 Tab Size: 4 UTF-8 LF PHP
27°C Haze
Search
ENG IN
23:19 24-05-2023

Phishing Attacks

Phishing attacks are a type of cyber-attack where individuals or associations essay to deceive others into revealing sensitive information similar as watchwords, credit card figures, or details. These attacks generally do through dispatch, instant messaging, or fraudulent websites that mimic licit sources. Then is a brief overview of how phishing attacks generally work Deceptive communication Phishers shoot out emails or dispatches that appear to be from a secure source, like a well-known company, a fiscal institution, or a government agency. They frequently use tactics like creating a sense of urgency, offering prices, or pretending to address a security issue. Social engineering Phishing attacks calculate social engineering ways to manipulate victims. They might use cerebral tricks to move individuals to click on vicious links, download attachments, or give sensitive information. Fake websites Phishers may produce fraudulent websites that imitate the look and sense of licit bones, like online banking doors

or login runners. These fake websites are designed to trick druggies into entering their credentials or other information. Information theft When victims interact with deceptive communication or fake websites, their sensitive information is captured by the phishers. This information can also be used for colorful vicious purposes, including identity theft, fiscal fraud, or unauthorized access to accounts. To cover yourself from phishing attacks, consider the following measures Be conservative of emails and dispatches Be cautious of unasked emails or d patches, especially those requesting information or fish l details. Double-check the lender's dispatch address and look for signs of suspicious exertion or inconsistencies. corroborate website authenticity Before entering sensitive information on a website, ensure it is it. Check the URL for any variations or misspellings and look for secure connections (HTTPS//) and security pointers suchlike padlock icons. Avoid clicking on suspicious links and hangover links in emails or dispatches to exercise the URL before clicking. Avoid clicking on links that feel suspicious or come from untrusted sources. rather, manually class the website's address in your cyber surfed. Keep software up to date Regularly modernize your operating system, web cybersurfed, and security software to cover against known vulnerabilities that phishers may exploit. Educate yourself and others Stay informed about common phishing ways and partake this knowledge with musketeers, family, and associates. Be conservative when participating information online or over the phone.


Phisher
4a.Credentials
5a.Profit
Hacker Redirected Website
1.Phishing Email
0.Upload Phishing Kit
3.Credentials
4b.Credentials
5b.Profit
2.Credentials
Hacker Site
Victim
ID Theft

Research Paper

Abstract:

The rise of phishing websites poses a significant threat to online users, particularly those seeking to purchase drugs, as they are vulnerable to deceptive practices aimed at obtaining sensitive information. To address this issue, a PHP-based phishing website detection system can be developed. This research paper explores the design and implementation of such a tool, emphasizing its ability to analyze URLs or domain names to identify potential phishing attempts. By utilizing various techniques, including domain name analysis and machine learning algorithms, the system provides a user-friendly interface and accurate detection capabilities tailored to drug users. The paper also discusses additional features such as user authentication, data storage, and integration with threat intelligence to enhance the overall effectiveness of the tool in protecting drug users from online fraud and scams.

Introduction:

The advent of the internet has revolutionized communication, information sharing, and access. However, it has also given rise to cybersecurity challenges, with phishing attacks being one of the most prevalent forms of online fraud. Phishing websites mimic legitimate platforms to deceive users into divulging sensitive data. This paper introduces a solution in the form of a PHP-based phishing website detection system, which utilizes advanced methods to effectively identify and mitigate these risks. Phishing has become a major concern for security experts, as creating convincing fake websites is relatively easy. While experts can identify such websites, not all users possess the necessary skills, making them vulnerable to phishing attacks. According to reports, businesses in the United States experience losses of up to $2

billion annually due to phishing attacks [1]. The third Microsoft Computing Safer Index Report estimated that the global impact of phishing could reach $5 billion [2].

2.1. Phishing Website Detection Methods: Comparing Domain Name Similarities: One of the key indicators of a phishing website is its domain name. By comparing it with established brands or checking for excessive hyphens and numbers, the detection system can identify suspicious URLs that imitate well-known platforms.

2.2. Length Analysis: Phishing websites often use lengthy domain names to confuse users. The system can estimate the length of a domain name and raise an alert for excessively long or suspicious URLs.

2.3. Machine Learning-Based Content Analysis: To enhance detection sensitivity, the system can utilize machine learning algorithms to analyze website content. By training the model on a large dataset of known phishing websites, it can identify patterns, keywords, and malicious scripts indicative of a phishing attempt [3]. The following criteria can be used as indicators:

- Presence of an IP address in the URL.
- Presence of the "@" symbol in the URL.
- Number of dots in the hostname.
- Prefix or suffix separated by a dash in the domain.
- URL redirection.
- Information submission to a specific destination.
- Usage of URL shortening services (e.g., "Tiny URL").
- Length of the hostname.
- Presence of sensitive words in the URL.

User-Friendly Interface:

The phishing website detection system is designed with a user-friendly interface, allowing users to easily input URLs or domain names for analysis. The tool provides clear and concise results, indicating whether a website is likely a phishing attempt, empowering users to make informed decisions while browsing.

Additional Features:

• User Authentication and Authorization: To enhance security, the system can incorporate user authentication and authorization mechanisms, ensuring only trusted users can access the tool.

• Data Storage and Retrieval: The tool can store analyzed URLs and their corresponding results in a database, facilitating future reference and analysis. This enables the system to improve its detection sensitivity over time.

• User Education: Educating users about phishing techniques and prevention strategies is crucial. The system can provide educational

5. Conclusion

A phishing website sensor using PHP offers an effective means of securing druggies against online swindles and fraud. By assaying sphere names, employing machine literacy ways, and incorporating fresh features like stoner authentication and trouble intelligence integration, the sensor can directly identify implicit phishing attempts. This stoner-friendly tool empowers individuals to make informed opinions while navigating the online geography, thereby mollifying the dangerous consequences of phishing attacks. As cybersecurity continues to evolve, the phishing website sensor serves as a pivotal defense medium in guarding druggies against online fraud.

6. Reference

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

[2] https://resources.infosecinstitute.com/category/enterprise/phishing/the-phishing-landscape/phishing-data-attack-statistics/#gref

[3] www.researchgate.net

[4] Mohammad R., Tabatha F. McCluskey L., (2015) Phishing websites dataset. Available: https://archive.ics.uci.edu/ml/datasets/Phishing+Websites Accessed January 2016.

[5] www.alexa.com

[6] www.phishtank.com

Phishing Website Detector

ORIGINALITY REPORT

5%	1%	1%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.tmu.ac.in Internet Source	1%
2	Submitted to Oklahoma Christian University Student Paper	1%
3	Submitted to University of Lancaster Student Paper	<1%
4	Submitted to University of North Florida Student Paper	<1%
5	Submitted to The University of Wolverhampton Student Paper	<1%
6	Submitted to Middlesex University Student Paper	<1%
7	Submitted to Academy of Allied Health & Science Student Paper	<1%
8	Submitted to Napier University Student Paper	<1%

9	Submitted to NALSAR University of Law Hyderabad Student Paper	<1%
10	Submitted to University of Salford Student Paper	<1%

Exclude quotes	On	Exclude matches	< 14 words
Exclude bibliography	On		

Phishing Website Detector

PAGE 1

S/V This subject and verb may not agree. Proofread the sentence to make sure the subject agrees with the verb.

S/V This subject and verb may not agree. Proofread the sentence to make sure the subject agrees with the verb.

PAGE 2

PAGE 3

Article Error You may need to use an article before this word.

PAGE 4

PAGE 5

Article Error You may need to use an article before this word.

PAGE 6

S/V This subject and verb may not agree. Proofread the sentence to make sure the subject agrees with the verb.

PAGE 7

PAGE 8

Article Error You may need to remove this article.

PAGE 9

Article Error You may need to use an article before this word. Consider using the article **the**.

Missing "," You may need to place a comma after this word.

PAGE 10

PAGE 11

P/V You have used the passive voice in this sentence. Depending upon what you wish to emphasize in the sentence, you may want to revise it using the active voice.

Article Error You may need to use an article before this word. Consider using the article **the**.

PAGE 12

Wrong Form You may have used the wrong form of this word.

PAGE 13

Article Error You may need to remove this article.

Article Error You may need to use an article before this word.

P/V You have used the passive voice in this sentence. Depending upon what you wish to emphasize in the sentence, you may want to revise it using the active voice.

PAGE 14

PAGE 15

Confused You have used **A** in this sentence. You may need to use **an** instead.

PAGE 16

Article Error You may need to use an article before this word.

PAGE 17

PAGE 18

PAGE 19

Missing "," You have a spelling or typing mistake that makes the sentence appear to have a comma error.

Article Error You may need to use an article before this word.

PAGE 20

Confused You have used **A** in this sentence. You may need to use **an** instead.

PAGE 21

P/V You have used the passive voice in this sentence. Depending upon what you wish to emphasize in the sentence, you may want to revise it using the active voice.

PAGE 22

Confused You have used **of** in this sentence. You may need to use **have** instead.

Article Error You may need to use an article before this word.

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

Proofread This part of the sentence contains a grammatical error or misspelled word that makes your meaning unclear.

Article Error You may need to use an article before this word.

PAGE 33

P/V You have used the passive voice in this sentence. Depending upon what you wish to emphasize in the sentence, you may want to revise it using the active voice.

Sentence Cap. Remember to capitalize the first word of each sentence.

Article Error You may need to use an article before this word.

Sentence Cap. Remember to capitalize the first word of each sentence.

PAGE 34

PAGE 35

Missing "," You may need to place a comma after this word.

Missing "," You may need to place a comma after this word.

PAGE 36

Article Error You may need to use an article before this word.

Article Error You may need to use an article before this word. Consider using the article **the**.

Frag. This sentence may be a fragment or may have incorrect punctuation. Proofread the sentence to be sure that it has correct punctuation and that it has an independent clause with a complete subject and predicate.

Article Error You may need to use an article before this word. Consider using the article **the**.

Frag. This sentence may be a fragment or may have incorrect punctuation. Proofread the sentence to be sure that it has correct punctuation and that it has an independent clause with a complete subject and predicate.

Frag. This sentence may be a fragment or may have incorrect punctuation. Proofread the sentence to be sure that it has correct punctuation and that it has an independent clause with a complete subject and predicate.

Frag. This sentence may be a fragment or may have incorrect punctuation. Proofread the sentence to be sure that it has correct punctuation and that it has an independent clause with a complete subject and predicate.

Frag. This sentence may be a fragment or may have incorrect punctuation. Proofread the sentence to be sure that it has correct punctuation and that it has an independent clause with a complete subject and predicate.

Article Error You may need to use an article before this word. Consider using the article **the**.

Frag. This sentence may be a fragment or may have incorrect punctuation. Proofread the sentence to be sure that it has correct punctuation and that it has an independent clause with a complete subject and predicate.

Article Error You may need to use an article before this word. Consider using the article **the**.

Frag. This sentence may be a fragment or may have incorrect punctuation. Proofread the sentence to be sure that it has correct punctuation and that it has an independent clause with a complete subject and predicate.

PAGE 37

Article Error You may need to use an article before this word. Consider using the article **the**.

PAGE 38
