A

PROJECT REPORT

ON

Phishing URL Detection

Submitted by

Patel Setukumar M. (19IT404)
Gandhi Kandarp A. (19IT410)
Panara Deep S. (19IT412)

For Partial Fulfillment of the Requirements for Bachelor of Technology in Information Technology

Guided by

Prof. Priyank N. Bhojak

April, 2023

Information Technology Department

Birla Vishvakarma Mahavidyalaya Engineering College

(An Autonomous Institution)

Vallabh Vidyanagar – 388120

Gujarat, INDIA

Birla Vishvakarma Mahavidyalaya Engineering College

(An Autonomous Institution)

Information Technology Department

AY: 2022-23, Semester VIII

CERTIFICATE

This is to certify that the project work entitled **Phishing URL Detection** has been successfully carried out by **Patel Setukumar M. (19IT404) , Gandhi Kandarp A. (19IT410) , Panara Deep S. (19IT412)** for the subject **Project II (4IT32)** during the academic year 2022-23, Semester-VIII for partial fulfilment of Bachelor of Technology in Information Technology. The work carried out during the semester is satisfactory.

Prof. Priyank N. Bhojak
IT Department
BVM

Dr. Keyur Brahmbhatt
Head, IT Department
BVM

Acknowledgement

We are extremely grateful to Dr. Indrajit Patel, Principal, Birla Vishvakarma Mahavidyalaya Engineering College and Dr. Keyur Brahmbhatt, Head of the Information Technology Department, for providing all the required resources for the successful completion of our project. Our heartfelt gratitude to our project guide Prof. Priyank N. Bhojak, Assistant Professor in Information Technology, for his valuable suggestions and guidance in the preparation of the progress report and implementation of project.

We express our thanks to project co-ordinator (Prof. Vishal A. Polara), and friends, for all the help and co-ordination extended in bringing out this project successfully in time.

We will be failing in duty if we do not acknowledge with grateful thanks to the authors of the references and other literatures referred to in this project.

Thanking You,

Setu Patel (19IT404)

Kandarp Gandhi (19IT410)

Deep Panara (19IT412)

Abstract

Nowadays use of internet is increased drastically because most of the people have smart phone , by which they can easily connected with the network. Although this makes easy our daily lives, it also brings many security breaches due to the anonymous structure of the Internet. Used antivirus programs and firewall systems can prevent most of the attacks. However, experienced attackers target on the weakness of the computer users by trying to phish them with bogus web pages. These pages imitate some popular banking, social media, e-commerce, etc. sites to steal some sensitive information such as, user-ids, passwords, bank account, credit card numbers, etc. Phishing detection is a challenging problem, and many different solutions are proposed in the market as a blacklist, rule-based detection, anomaly-based detection, etc.In this project, we proposed a machine learning-based phishing url detection system by using sixteen different feature to analyze the URLs and we use Kaggle dataset to train our model.We make website and chrome extension to check the url.

Table of Content

Chapter 1: Introduction

Chapter 2: Literature Review

Chapter 3: System Analysis & Design

List Of Figures

List Of Tables

List Of Abbreviations

SVM - Support Vector Machine

ANN - Artificial Neural Network

API - Application Programming Interface

URL - Universal Resource Locater

Chapter 1 : Introduction

1.1 Brief Overview :-

Phishing is the most commonly used social engineering and cyber attack. Through such attacks, the phisher targets naïve online users by tricking them into revealing confidential information, with the purpose of using it fraudulently. In order to avoid getting phished, users should have awareness of phishing websites.This System is machine learning based model.We use Kaggle as our dataset.We extract 16 different feature from those url and train our model.We use decision tree, svm and ann to train the model and we got maximum accuracy from ann. So, we decide to use ann as our machine learning algorithm,and make api of this model to use in our website. We make one website, with the help of that website user can check the url and get one alert box if url is phishing. On that web site, we also provoide different information about current cyber attacks, use of different tool and also different government website on which people report their complain about the cyber attack which they faced. We also make chrome extension.When user visit website at that time user can check website using that extension.

1.2 Objective :-

The main objective of this system to provide the way to check the URL and get the knowledge like the URL is safe or unsafe.This system provides the certain degree of trust to user to access the URL and protect user data by not visiting phishing website.

1.3 Scope :-

It can be beneficial for user which uses internet as main source of getting knowledge because they visit different websites.It can be embedded in browser as extension.

1.4 Modules :-

Data Gathering

Data Pre-processing

Feature Extraction

Model Generation

Web App Development

Chrome extension

1.5 Project Hardware And Software Requirements :-

1.5.1 Hardware:-

Client side:

Basic computer/laptop that can run browser.

Server Side:

Basic computer/laptop with Win10/Linux with integrated graphics

4 GB RAM

x86 64-bit CPU (Intel / AMD architecture)

1.5.2 Software :-

Visual Studio

Google colab

Postman

Pycharm

Bootstrap

Chapter 2: Literature Review

1) Detection of URL based Phishing Attacks using Machine Learning

Publisher : International Journal of Engineering Research & Technology

Author : Ms. Sophiya Shikalgar, Dr. S.D. Sawarkar, Mrs. Swati Narwane

In their research work they have seen that existing system gives less accuracy so we proposed a new phishing method that employs URL based features and also we generatedclassifiers through several machine learning algorithms and they have used different ML Classifier and like, XG Boost, SVM classifier, Naïve Bayes Classifier and proposed that their best model is XG Boost model and it achieved the accuracy 86.3 % . The proposed technique is much more secured as it detects new and previous phishing sites. They have used various basic features, bag of words, fussy features, distance-based features to train their model.

2) Phishing Website Detection using Machine Learning

Publisher : International Journal of Scientific Research in Computer Science

Author : Purvi Pujara, M.B. Chaudhari

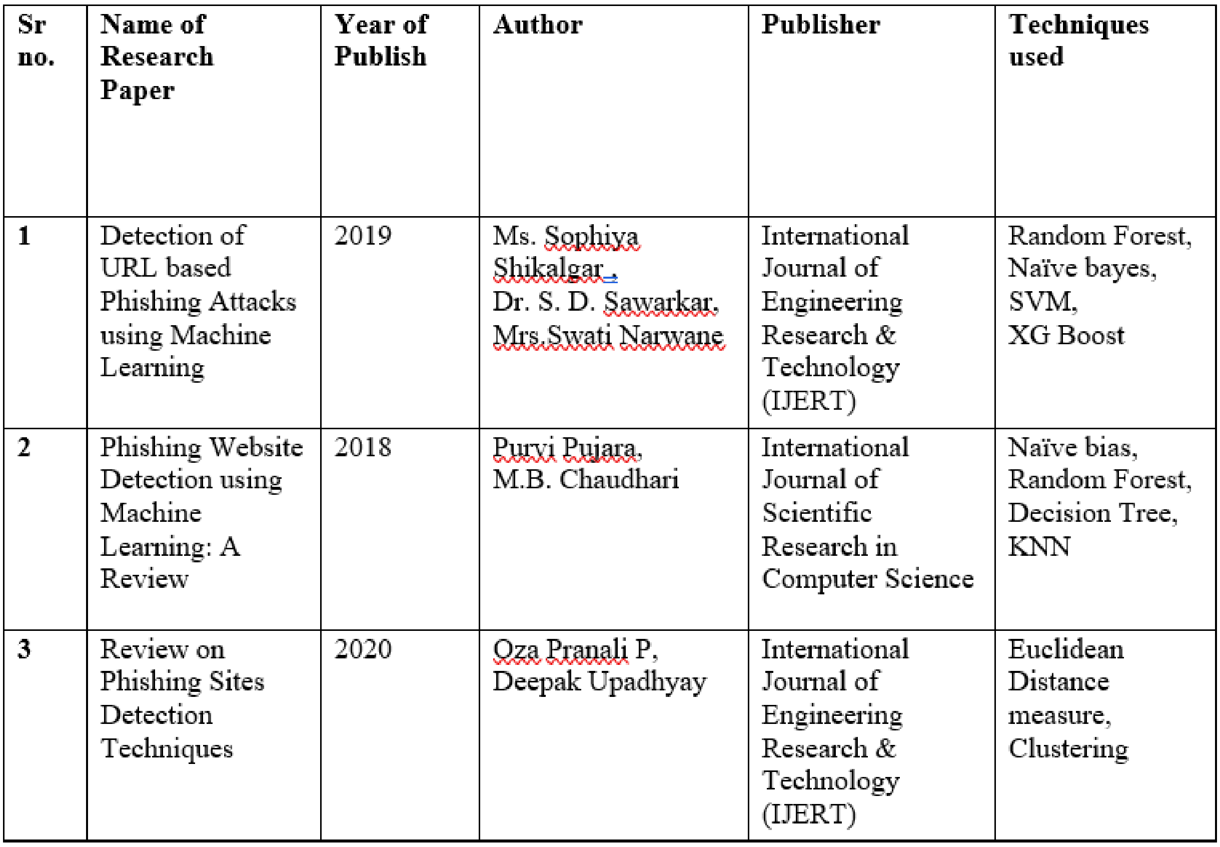
This study shows how phishing can be harmful in different ways. Phishing is a way to obtain user's private information via email or website. As usage of internet is very vast, almost all things are available online now it is either about shopping cloths, electronic gadgets, crockery or to payment of mobile, TV & electricity bill. Rather than standing outin line for hours, people are being aware of using online method. Due to this phisher has wide scope to implement phishing scam. As there is lot of research work done in this area,there is not any single technique, which is enough to detect all types of phishing attack. As technology increases, phishing attackers using new methods day by day. This enables us to find effective classifier to detection of phishing. we performed detailed literature surveyabout phishing website detection. According to this, we can say tree-based classifiers in machine learning approach is best suitable than other.

3) Review on Phishing Sites Detection Techniques

Publisher : International Journal of Engineering Research & Technology

Author : Oza Pranali P, Deepak Upadhyay

In this paper, we described literature survey about phishing website detection. Phishing websites are short-lived, and thousands of fake websites are generated every day. Therefore, there is requirement of real-time, fast and intelligent phishing detection solution. According to this, Machine learning is efficient technique to detect phishing. More featurescan be added to improve the accuracy of the proposed phishing detection system.



Sr no.	Name of Research Paper	Year of Publish	Author	Publisher	Techniques used
1	Detection of URL based Phishing Attacks using Machine Learning	2019	Ms. Sophiya Shikalgar, Dr. S. D. Sawarkar, Mrs.Swati Narwane	International Journal of Engineering Research & Technology (IJERT)	Random Forest, Naïve bayes, SVM, XG Boost
2	Phishing Website Detection using Machine Learning: A Review	2018	Purvi Pujara, M.B. Chaudhari	International Journal of Scientific Research in Computer Science	Naïve bias, Random Forest, Decision Tree, KNN
3	Review on Phishing Sites Detection Techniques	2020	Oza Pranali P, Deepak Upadhyay	International Journal of Engineering Research & Technology (IJERT)	Euclidean Distance measure, Clustering

Table 2.1. Literature Review

Chapter 3 : System Analysis & Design

3.1 Comparison Of Existing Applications With This Project With Merits And Demerits:-

Merit -

- Here, In this project there is custom web application for user to check web url and we use ann to train our model.

We also make chrome extension.

Demerit -

- Currently our model is trained on limited data due to low end device laptop which can be increased by using high end machines to train data.

3.2 Project Feasibility Study :-

3.2.1 Technical Feasibility :-

The technical issue usually raised during the feasibility stage of the investigation includes the following :

- Does the necessary technology exist to do what is suggested?
  Yes, We can build this system using existing technologies.

- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
  Yes, System will provide adequate response to inquiries as quickly as possible.

- Can the system be upgraded if developed?
  Yes, System can be upgraded after it is developed.

- Are there technical guarantees of accuracy, reliability, ease of access and data security?
  Yes, There are technical guarantees of accuracy, reliability, ease of access and data security.

3.2.2 Operational Feasibility :-

Some of the important issues raised are to test the operational feasibility of a project includes the following :

- Will the system be used and work properly if it is being developed and implemented?
  Yes, System will work properly if it is being developed and implemented.

- Is there sufficient support for the management from the users?
  Yes, There is sufficient support for the management from the users.

3.2.3 Economic Feasibility :-

Economic Feasibility looks for the financial aspects of the system. Economic Feasibility concern with the returns from the investment in a system. It determines whether it is worthwhile to invest in that proposed system.

3.2.4 Software Development Life Cycle :-

We used the concept of water fall model. Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

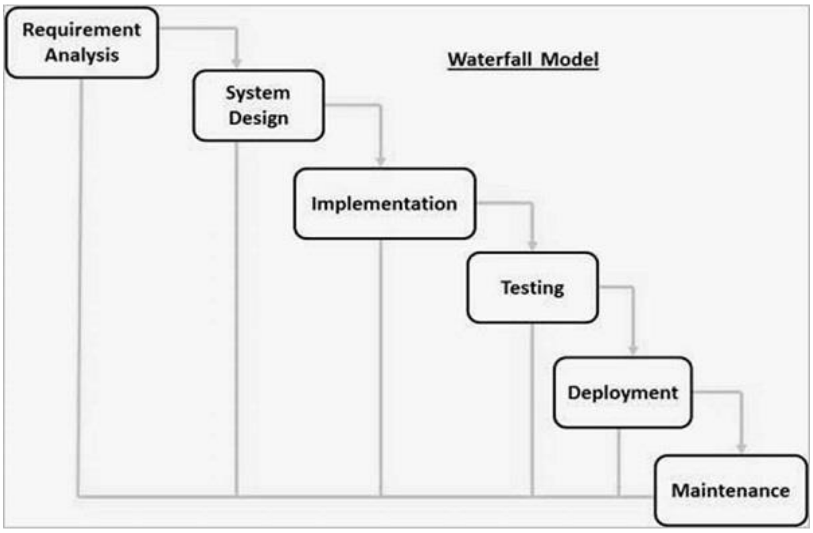


Figure 1: Water Fall Model

3.3 Project Timeline chart :-

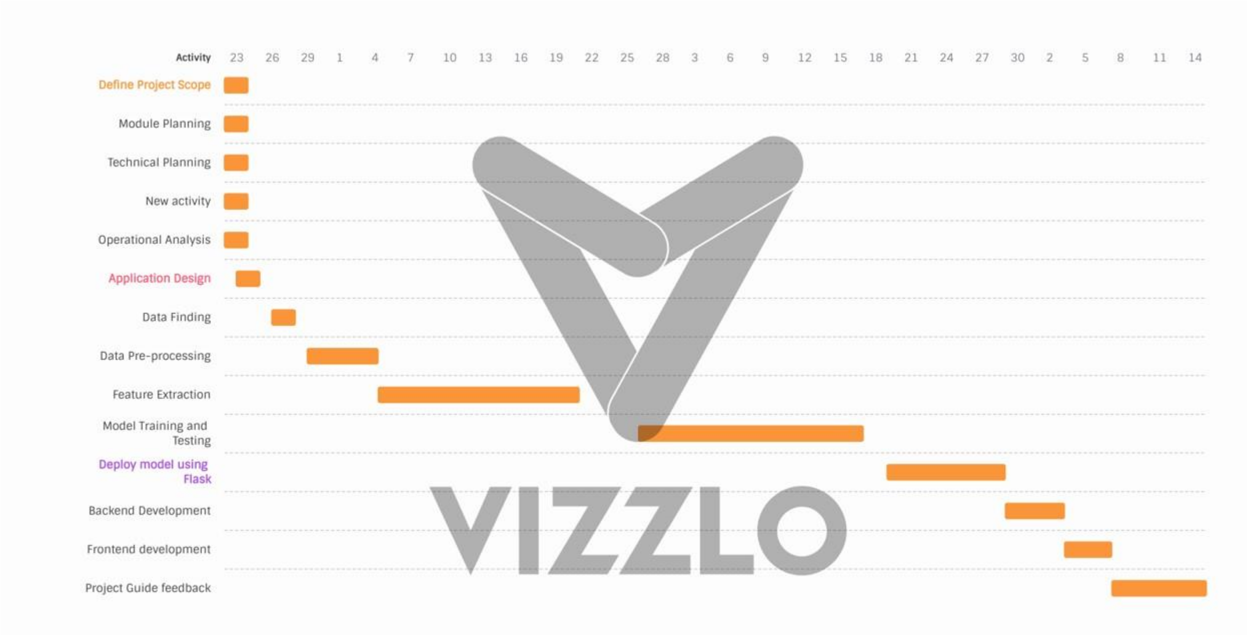


Figure 2: Time Line Chart

- We started our project development in January and we decided our project between 17th July and 18th January.
- In the very next week, we have planned about the modules and feasibility study for our project.
- in the end of the January month, we have worked on scope and technical planning.
- In the very first week of February , we started to work on the data pre-processing
- By the end of the February month, we extract features.
- By the end of the February month, we started model training and completed in the middle of march month.
- After that we deploy our model using flask.
- In the end of march, we develop backend and frontend of our website. And also we develop chrome extension.

3.4 Detailed Modules Description :-

3.4.1 Data gathering :-

- We will gather dataset from Kaggle. The data we collect to develop practical ML solutions, it must be collected and stored in a way that makes sense for the business problem at hand.
- Collecting data allows you to capture a record of past events so that we can use data analysis to find recurring patterns.

3.4.2 Data Pre-processing :-

- It is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.
- Data pre-processing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.
- To execute this, we will first get the dataset then import libraries and encode categorial data then split into training and test data.
- Handling missing values in dataset.
- Removing constant features, outliers, etc.
- Min-Max Scaler shrinks the data within the given range, usually of 0 to 1. It transforms data by scaling features to a given range. It scales the values to a specific value range without changing the shape of the original distribution.

3.4.3 Feature extraction :-

- **Address Bar Based Features**:

  - Domain of URL

    Here, we are just extracting the domain present in the URL. This feature doesn't have much significance in the training. May even be dropped while training the model.

  - IP Address in URL

    Checks for the presence of IP address in the URL. URLs may have IP address instead of domain name. If an IP address is used as an alternative of the domain name in the URL, we can be sure that someone is trying to steal personal information with this URL.

  - "@" Symbol in URL

    Checks for the presence of '@' symbol in the URL. Using “@” symbol in the URL leads the browser to ignore everything preceding the “@” symbol and the real address often follows the “@” symbol.

  - Length of URL

    Computes the length of the URL. Phishers can use long URL to hide the doubtful part in the address bar. In this project, if the length of the URL is greater than or equal 54 characters then the URL classified as phishing otherwise legitimate.

  - Depth of URL

    Computes the depth of the URL. This feature calculates the number of sub pages in the given url based on the '/'.

  - Redirection "//" in URL

    Checks the presence of "//" in the URL. The existence of “//” within the URL path means that the user will be redirected to another website. The location of the “//” in URL is computed. We find that if the URL starts with “HTTP”, that means the “//” should appear in the sixth position. However, if the URL employs “HTTPS” then the “//” should appear in seventh position.

- "http/https" in Domain name

  Checks for the presence of "http/https" in the domain part of the URL. The phishers may add the “HTTPS” token to the domain part of a URL in order to trick users.

- Using URL Shortening Services “TinyURL”

  URL shortening is a method on the “World Wide Web” in which a URL may be made considerably smaller in length and still lead to the required webpage. This is accomplished by means of an “HTTP Redirect” on a domain name that is short, which links to the webpage that has a long URL.

- Prefix or Suffix "-" in Domain

  Checking the presence of '-' in the domain part of URL. The dash symbol is rarely used in legitimate URLs. Phishers tend to add prefixes or suffixes separated by (-) to the domain name so that users feel that they are dealing with a legitimate webpage.

- **Domain Based Features:**

  - DNS Record

    For phishing websites, either the claimed identity is not recognized by the WHOIS database or no records founded for the hostname. If the DNS record is empty or not found then, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

  - Website Traffic

    This feature measures the popularity of the website by determining the number of visitors and the number of pages they visit.

  - Age of Domain

This feature can be extracted from WHOIS database. Most phishing websites live for a short period of time. The minimum age of the legitimate domain is considered to be 12 months for this project.

- End Period of Domain

  This feature can be extracted from WHOIS database. For this feature, the remaining domain time is calculated by finding the different between expiration time & current time.

- **HTML and JavaScript Based Features:**

  - IFrame Redirection

    IFrame is an HTML tag used to display an additional webpage into one that is currently shown. Phishers can make use of the “iframe” tag and make it invisible i.e. without frame borders. In this regard, phishers make use of the “frameBorder” attribute which causes the browser to render a visual delineation.

  - Status Bar Customization

    Phishers may use JavaScript to show a fake URL in the status bar to users. To extract this feature, we must dig-out the webpage source code, particularly the “onMouseOver” event, and check if it makes any changes on the status bar

  - Disabling Right Click

    Phishers use JavaScript to disable the right-click function, so that users cannot view and save the webpage source code. This feature is treated exactly as “Using onMouseOver to hide the Link”. Nonetheless, for this feature, we will search for event “event.button==2” in the webpage source code and check if the right click is disabled.

- Website Forwarding

  The fine line that distinguishes phishing websites from legitimate ones is how many times a website has been redirected. In our dataset, we find that legitimate websites have been redirected one time max. On the other hand, phishing websites containing this feature have been redirected at least 4 times.

3.4.4 Model generation :-

- In order to incorporate machine learning capabilities into a computer application, we generate a model to try and test many different algorithms, tools, and parameters to get the desired result with accuracy.

3.4.5 Web app development and extension :-

- In this module we will make web app and chrome extension.
- There is one textbox in web app.user have to enter the url and then click the submit button to check the url.
- In chrome extension, when user visit web site at that time user have click the button to check the url.

3.5 Project SRS :-

3.5.1 Use Case Diagrams :-

Use case diagrams are a common way to communicate the major functions of a software system. A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

Use cases are nothing but the system functionalities written in an organized manner. Now another thing which is relevant to the use cases are the actors. Actors can be defined as something that interacts with the system.

So in brief, the purposes of use case diagrams can be as follows:

- Used to gather requirements of a system.
- Used to get an outside view of a system.
- Identify external and internal factors influencing the system.
- Show the interacting among the requirements are actors.

Symbols used in Use Case diagram:

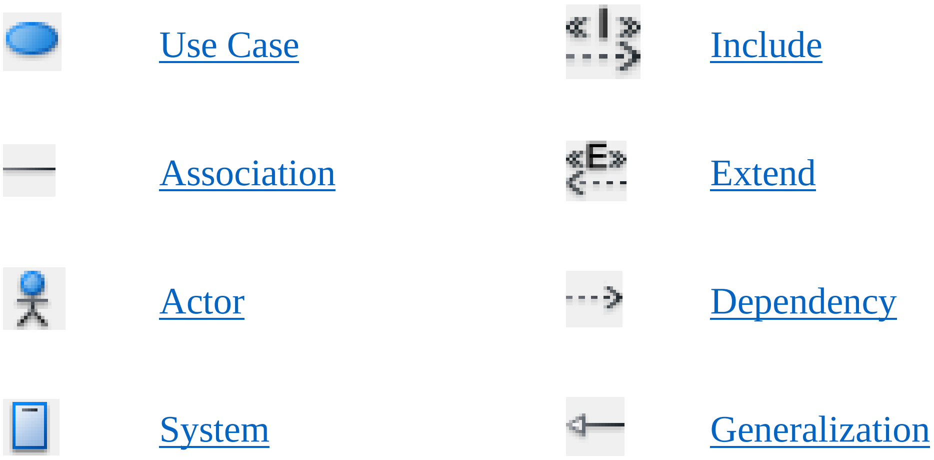


Figure 3: Symbols of Use case diagram

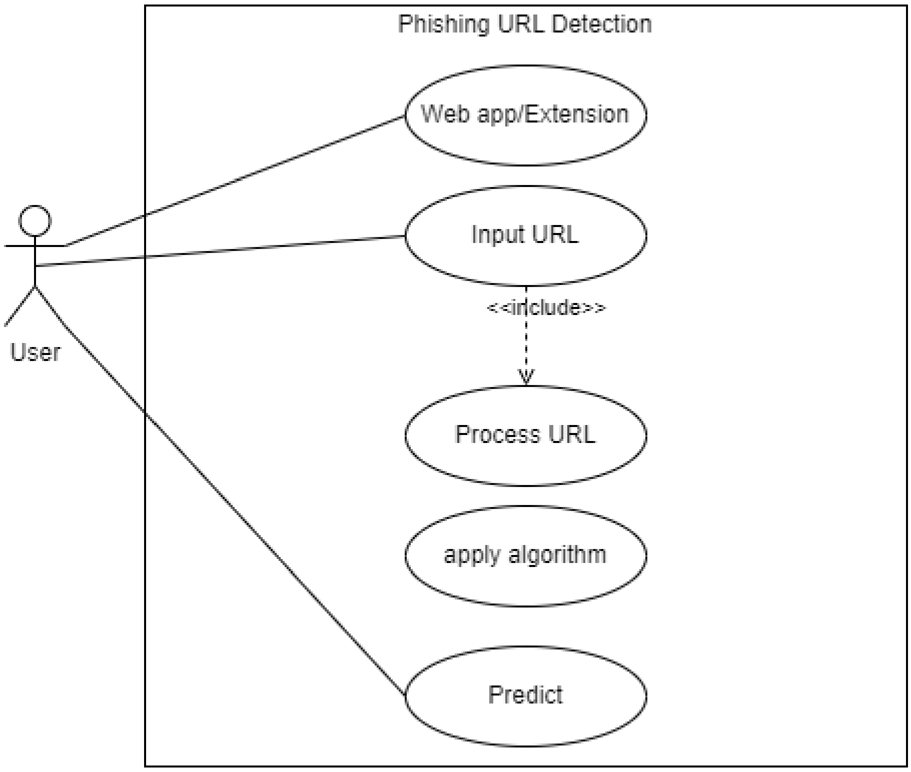


Figure 4: Use case diagram

3.5.2 Data Flow Diagrams :-

DFD provides the functional overview of a system. The graphical representation easily overcomes any gap between user and system analyst and analyst and system designer in understanding a system. Starting from an overview of the system it explores detailed design of a system through a hierarchy. DFD shows the external entities from which data flows into the process and also the other flows of data within a system. It also includes the transformations of data flow by the process and the data stores to read or write a data.

Symbols used in Data Flow diagram :-

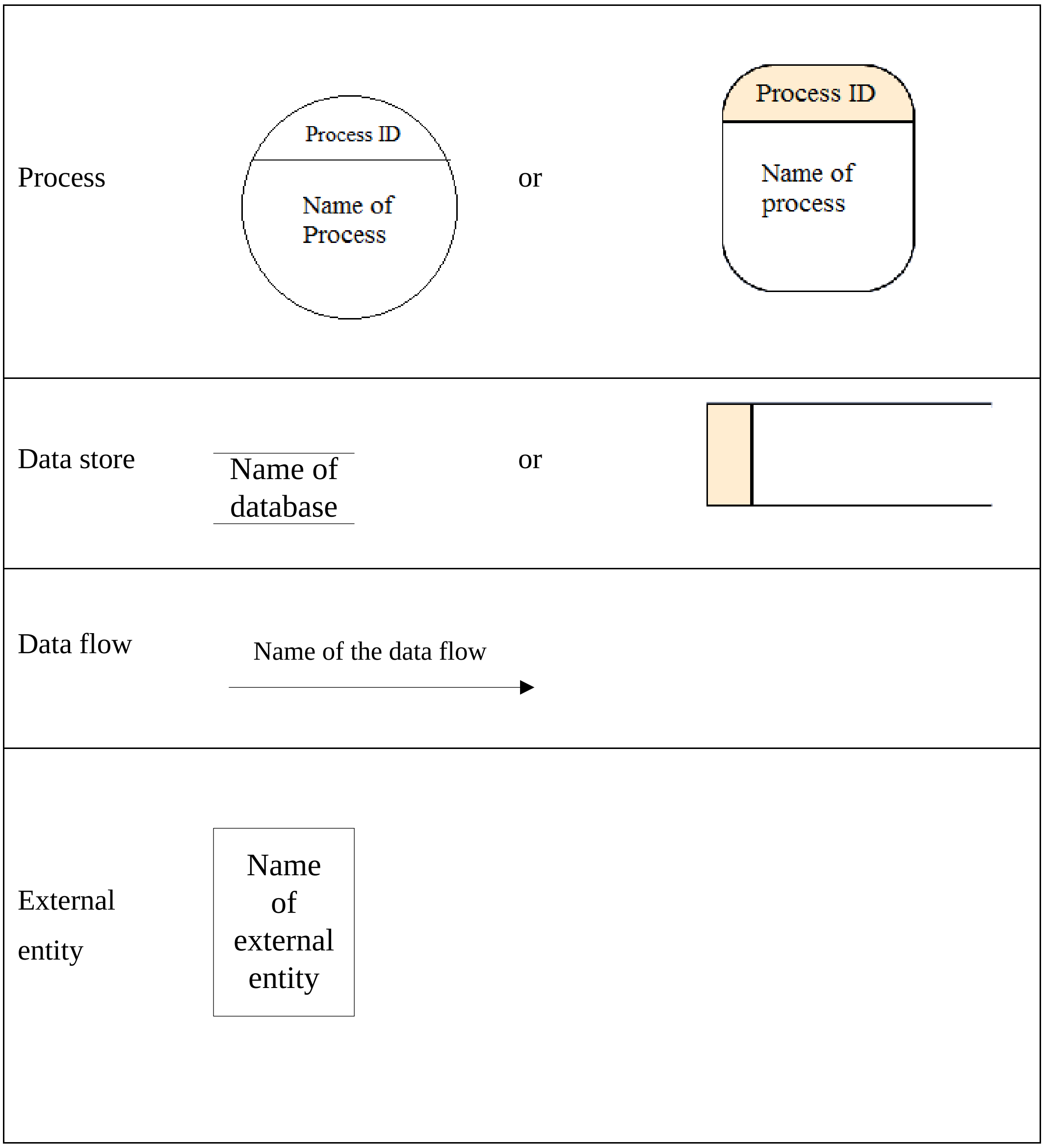


Figure 5: Symbols of DFD

Level 0:

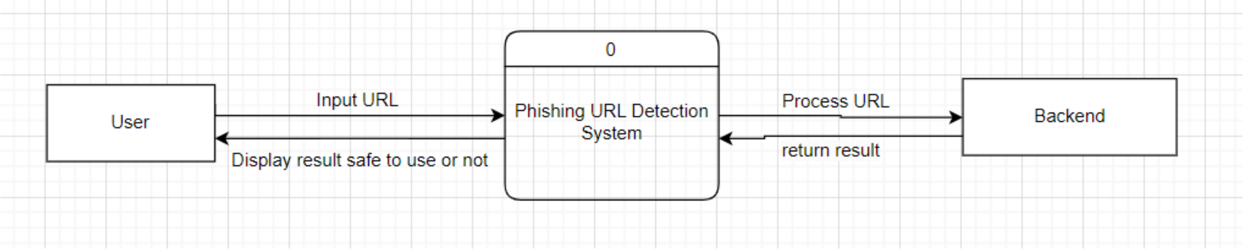

0
User
Input URL
Display result safe to use or not
Phishing URL Detection System
Process URL
return result
Backend

Figure 6: Data Flow Diagram(level 0)

Level 1:

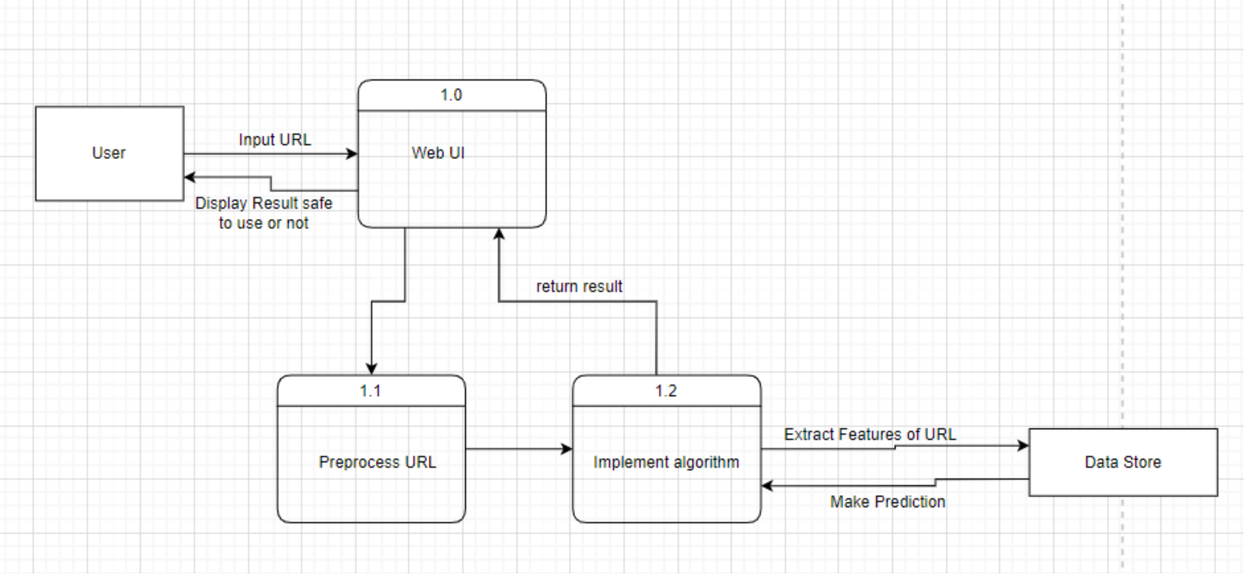


Figure 7: Data Flow Diagram (level 1)

3.5.4 Event Trace Diagram :-

A sequence diagram is an interaction diagram that emphasizes the time ordering of messages. It shows a set of objects and the messages sent and received by those objects.

Graphically, a sequence diagram is a table that shows objects arranged along the X axis and messages, ordered in increasing time, along the Y axis.

An object in a sequence diagram is rendered as a box with a dashed line descending from it. The line is called the object lifeline, and it represents the existence of an object over a period of time**.**

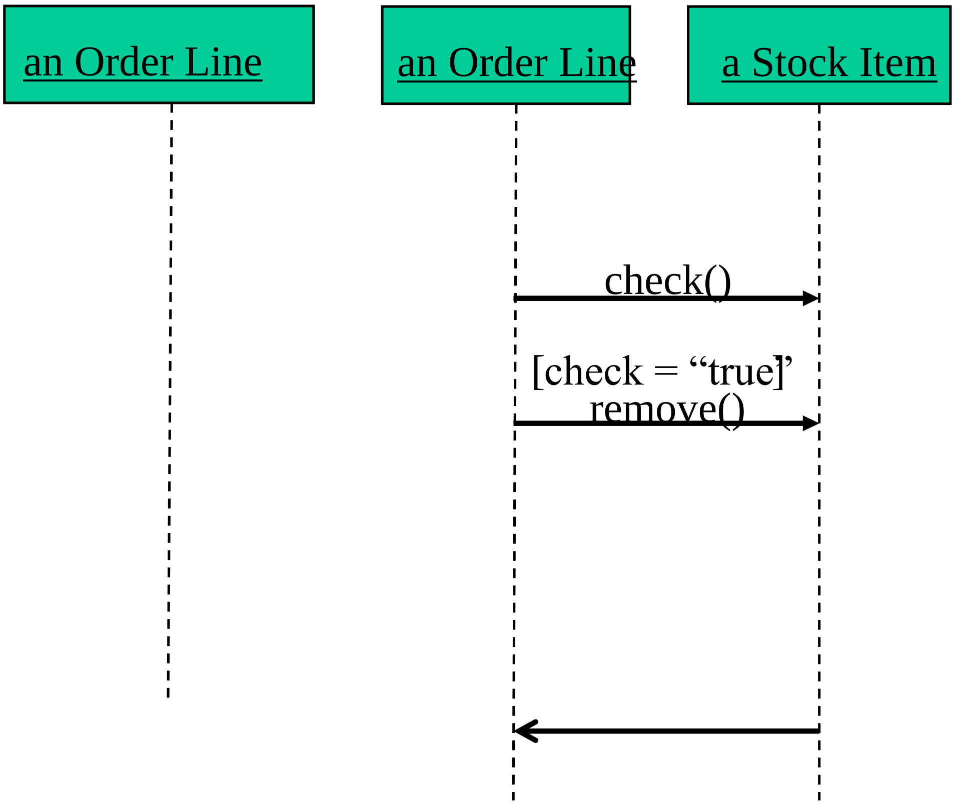


Figure 8: Symbols of Event Trace diagram

Messages are rendered as horizontal arrows being passed from object to object as time advances down the object lifelines. Conditions (such as [check = “true”]) indicate when a message gets passed.

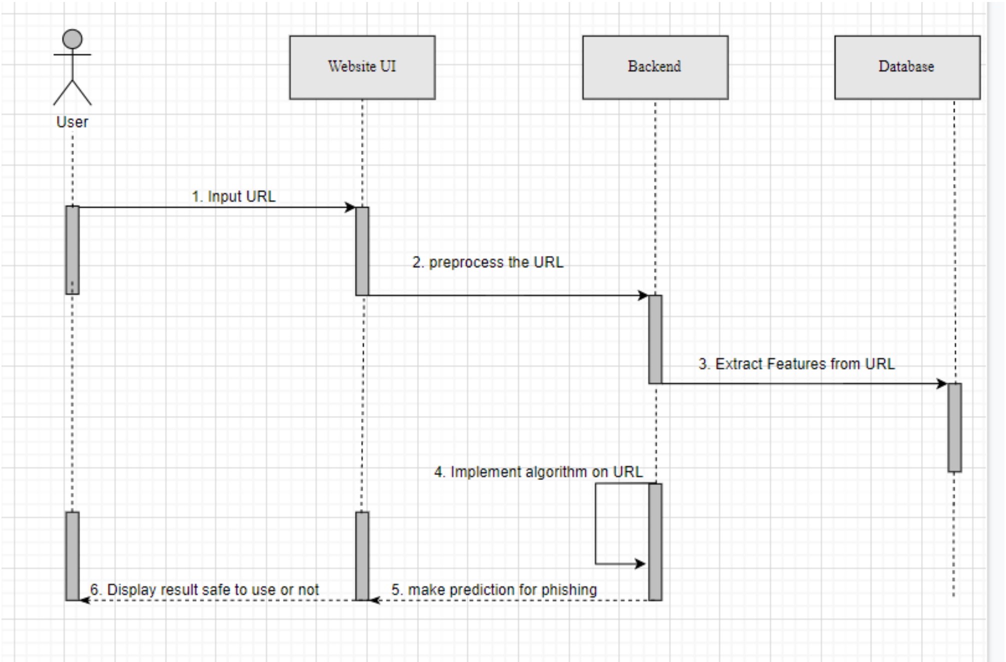

User
Website UI
Backend
Database
1. Input URL
2. preprocess the URL
3. Extract Features from URL
4. Implement algorithm on URL
5. make prediction for phishing
6. Display result safe to use or not

Figure 9: Event Trace Diagram

3.5.5 State diagram :-

“The state diagram describes the dynamic behavior of objects over time by modeling the life cycles of objects of each class.

Each object is treated as an isolated entity that communicates with the rest of the world by detecting events and responding to them.

Events represent the kinds of changes that objects can detect... Anything that can affect an object can be characterized as an event.”

An object must be in some specific state at any given time during its life cycle.

An object transitions from one state to another as the result of some event that affects it.

There can be only one start state in a state diagram, but there may be many intermediate and final states.

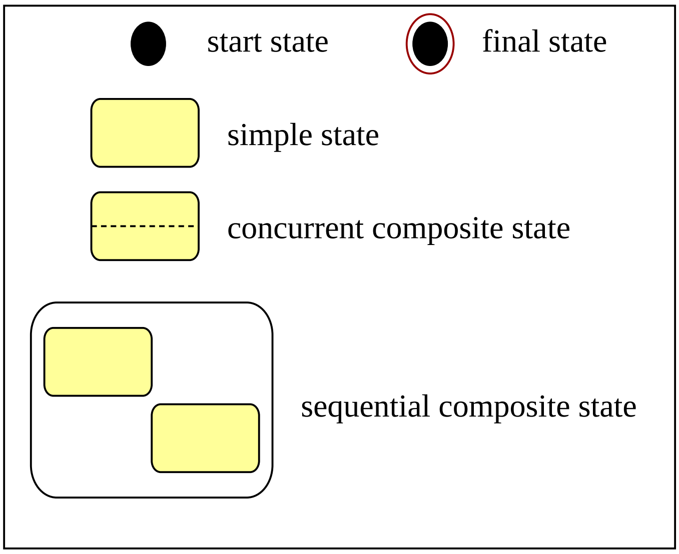


Figure 10: Symbols of State diagram

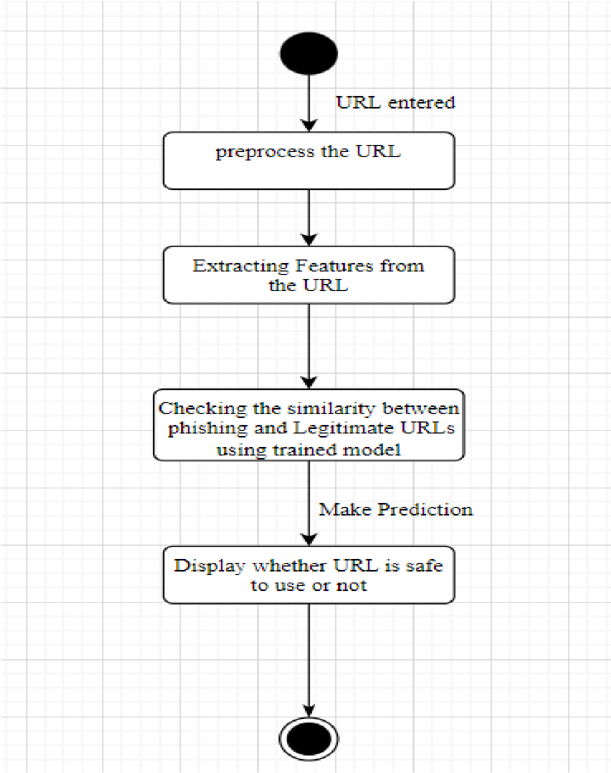

URL entered
preprocess the URL
Extracting Features from the URL
Checking the similarity between phishing and Legitimate URLs using trained model
Make Prediction
Display whether URL is safe to use or not

Figure 11: State Diagram

3.5.6 System Flow Diagram:-

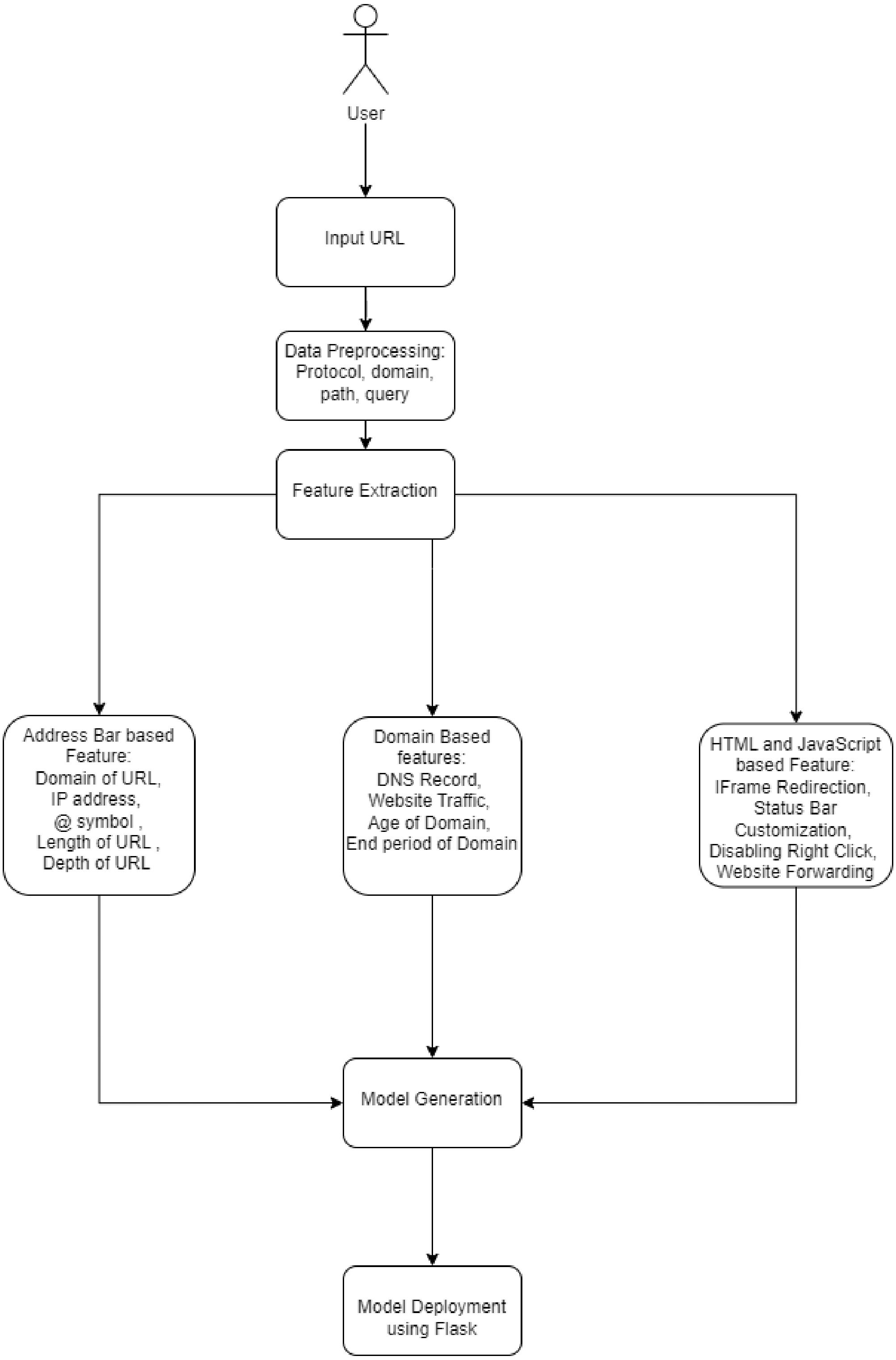


Figure 12: system flow diagram

Chapter 4 : Implementation and Testing

4.1 User Interface and Snapshot :-

4.1.1 Frontend :-

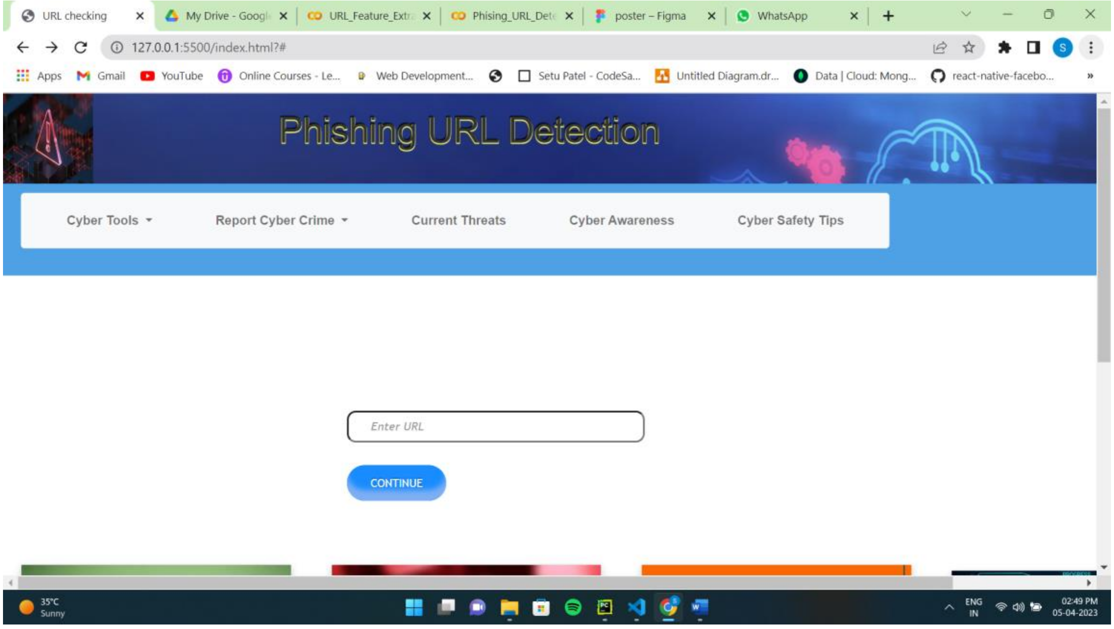
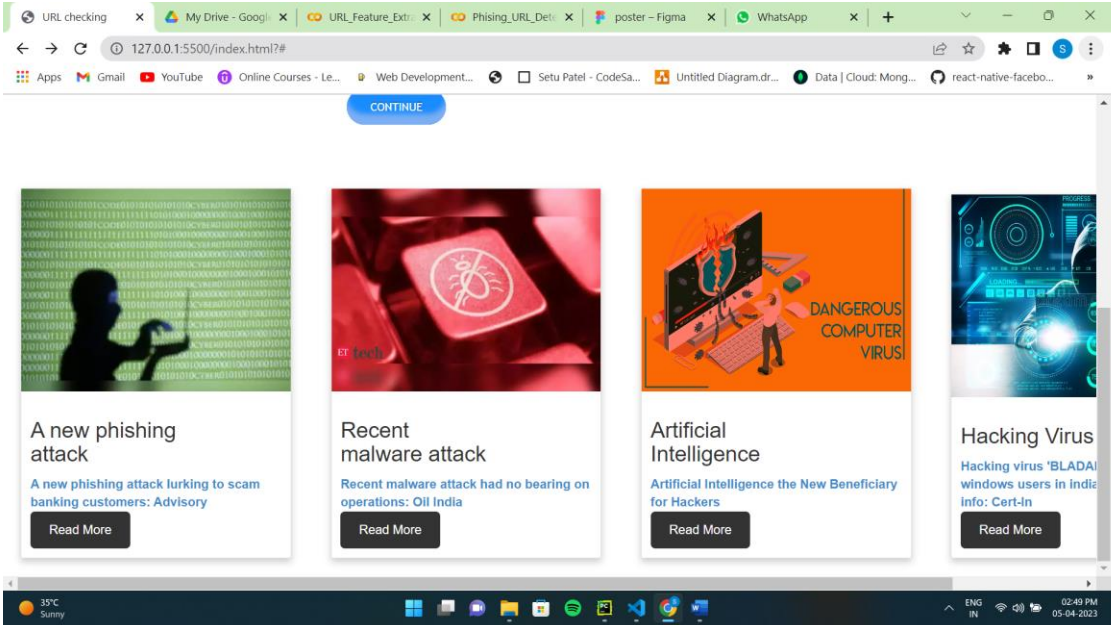


Figure 13: web UI (1)



A new phishing attack

A new phishing attack lurking to scam banking customers: Advisory

Recent malware attack

Recent malware attack had no bearing on operations: Oil India

Artificial Intelligence

Artificial Intelligence the New Beneficiary for Hackers

Hacking Virus

Hacking virus 'BLADAI windows users in india info: Cert-In

Figure 14: web UI (2)

4.1.2 Chrome Extension:-

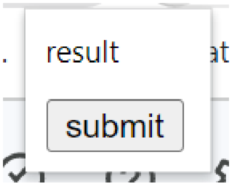
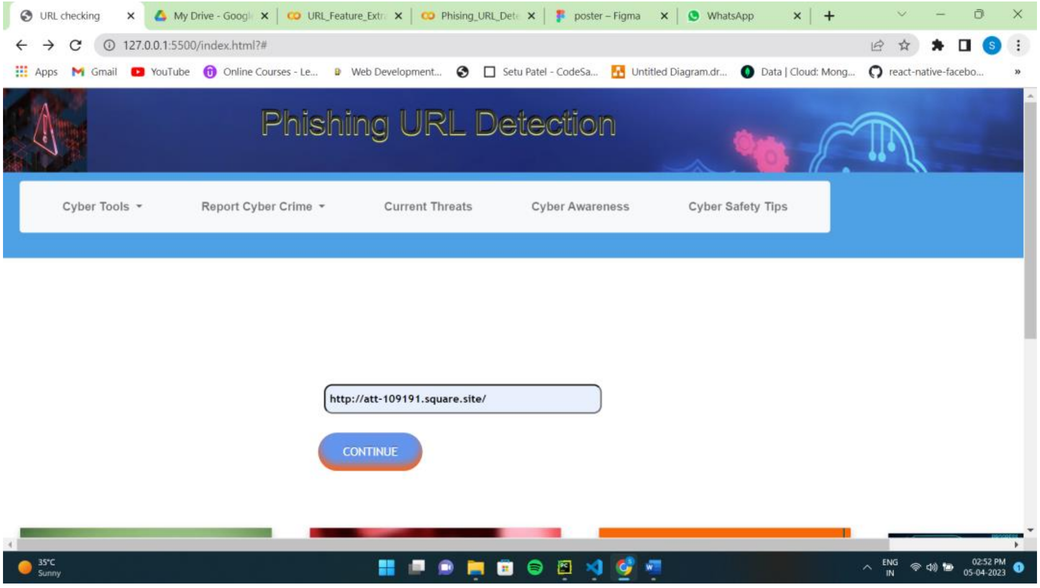


Figure 15: Chrome Extension

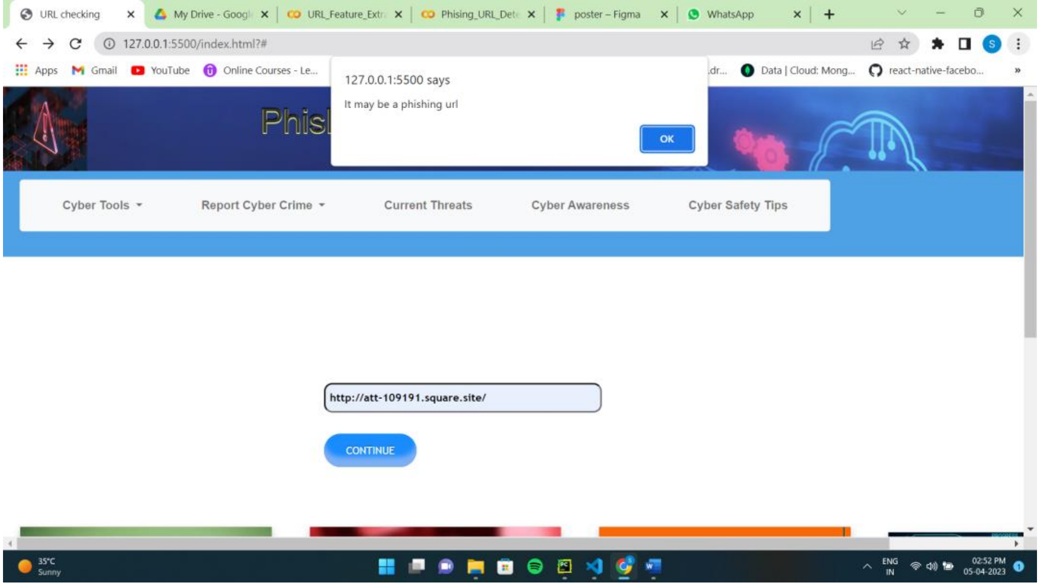
4.2: Testing using use cases:

4.2.1 User checks URL through web app :-



http://att-109191.square.site/

Figure 16: User input URL



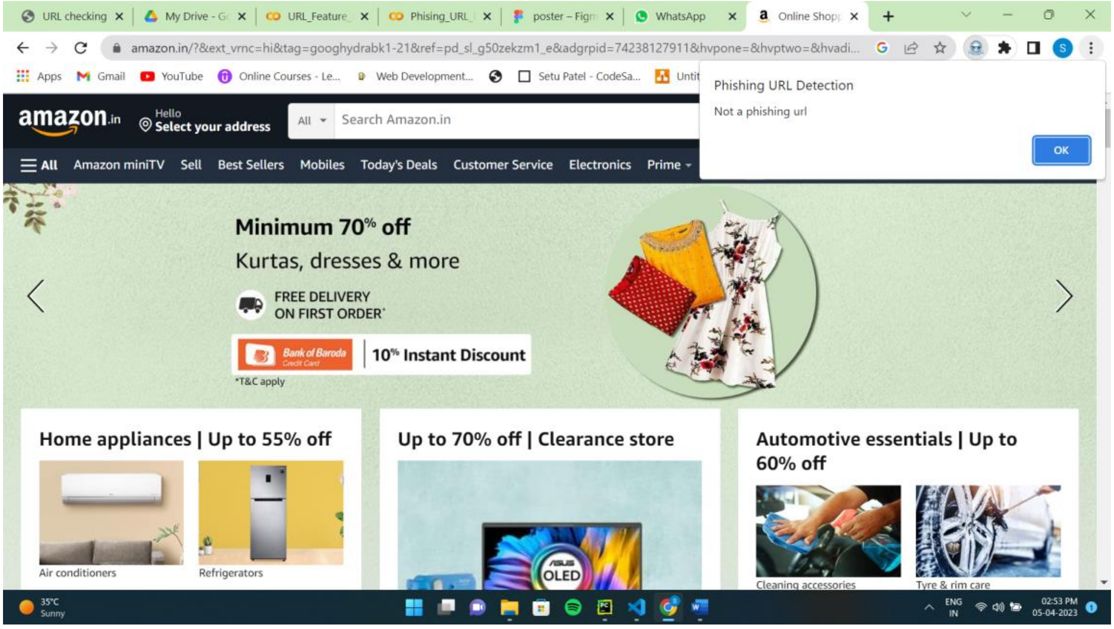
127.0.0.1:5500 says

It may be a phishing url

http://att-109191.square.site/

Figure 17: Gives prediction in alert box

4.2.2 User checks URL through chrome extension :-



Phishing URL Detection

Not a phishing url

OK

Figure 18: User check URL through extension

4.2.3: value loss and accuracy: -

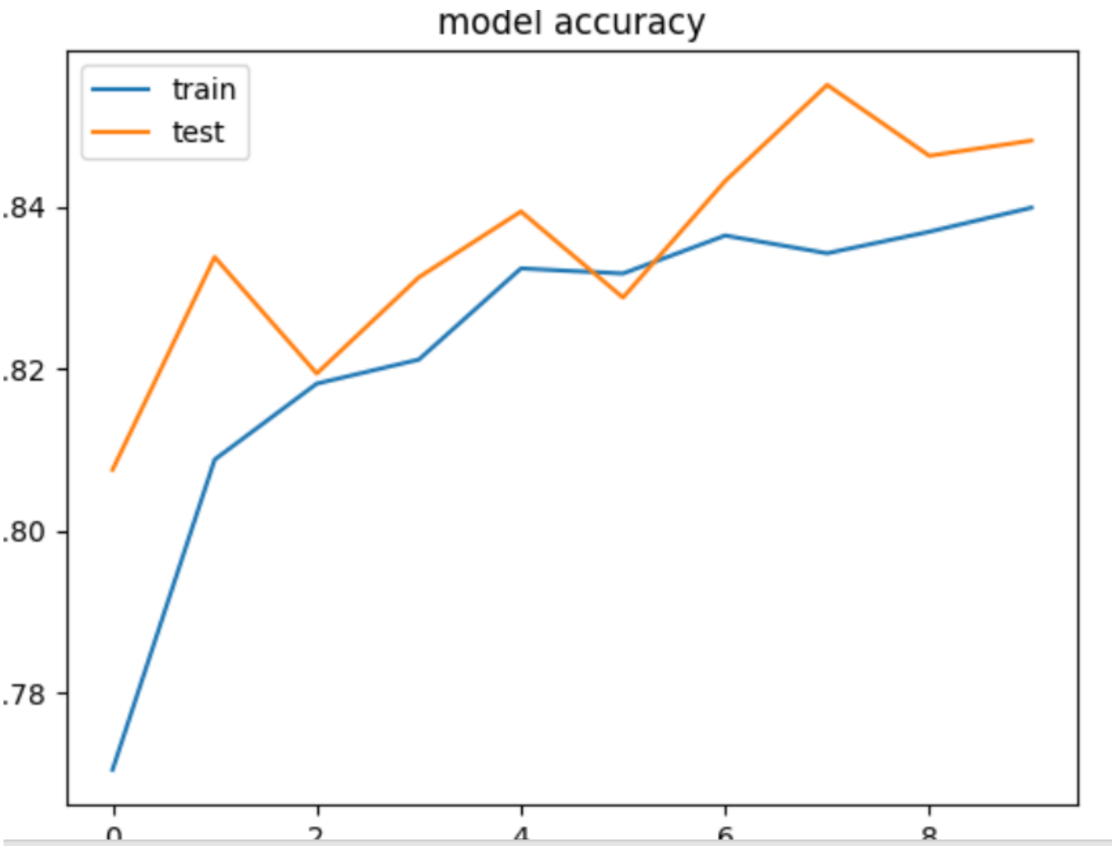


Figure 19: accuracy graph

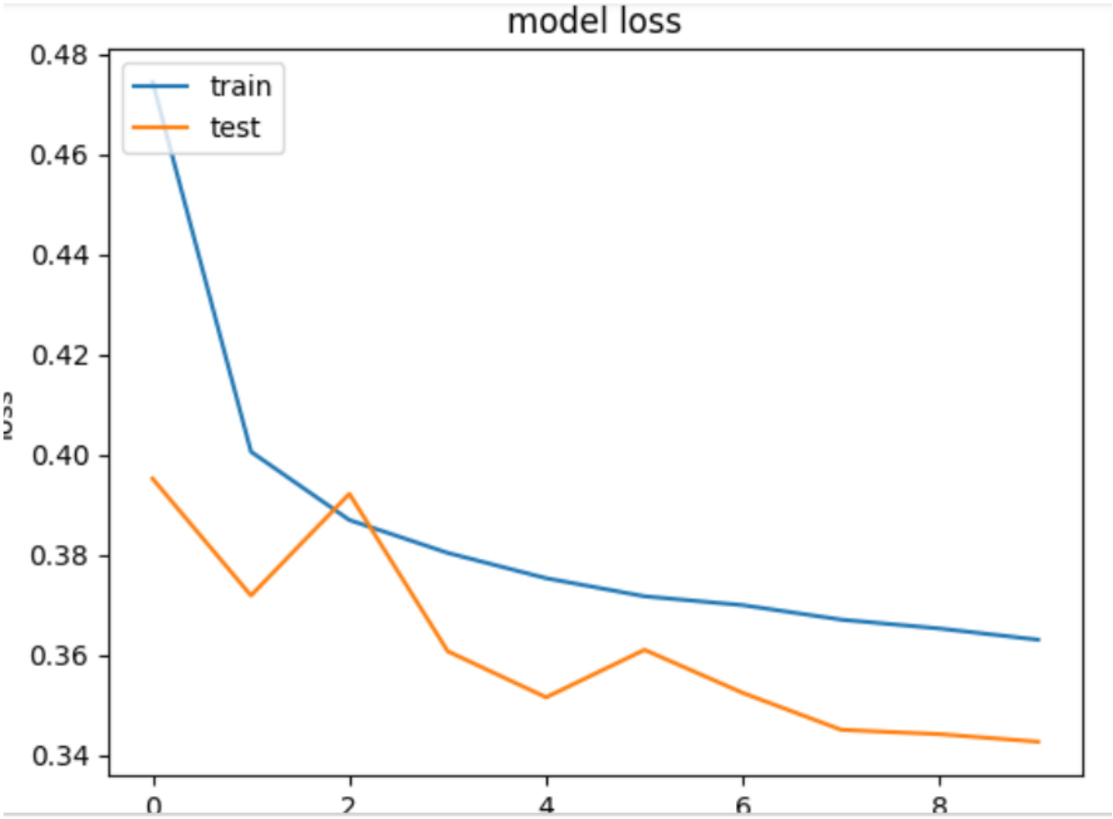


Figure 20: value loss graph

Chapter 5 : Conclusion & Future work

5.1 Conclusion :-

It is found that phishing attacks is very crucial and it is important for us to get a mechanism to detect it. As very important and personal information of the user can be leaked through phishing websites, it becomes more critical to take care of this issue. This problem can be easily solved by using any of the machine learning algorithm with the classifier. We already have classifiers which gives good prediction rate of the phishing beside , but after our survey that it will be better to use a hybrid approach for the prediction and further improve the accuracy prediction rate of phishing websites. We have seen that existing system gives less accuracy so we proposed a new phishing method that employs URL based features and also we generated classifiers through several machine learning algorithms. We have found that our system provides us with 80.75 % of accuracy for Decision Tree Classifier, 79.60% accuracy for SVM Classifier, 81.35 % accuracy for Random Forest Classifier and finally 83.40% of accuracy when using ANN. Hence we found that the best among all the above classifiers is ANN which shows maximum accuracy. The proposed technique is much more secured as it detects new and previous phishing sites.

5.2 Future work :-

In future if we get structured dataset of phishing, we can perform phishing detection muchfaster than any other technique. In future we can use a combination of any other two or more classifier to get maximum accuracy.

Chapter 6 : References

- [1]International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 8 Issue 11, November-2019. "Detection of URL based Phishing Attacks using Machine Learning"
- [2]International Journal of Engineering Research & Technology (IJERT), ISSN: 2456-3307, Vol. 3 Issue, 2019. "Phishing Website Detection using Machine Learning : A Review"
- [3]International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 9 Issue 04, April-2020. "Review on Phishing Sites Detection Techniques"

- https://medium.com/analytics-vidhya/phishing-url-detection-using-ml-4114d9930d61
- http://dataaspirant.com/2017/01/30/how-decision-treealgorithm-works/
- http://dataaspirant.com/2017/05/22/random-forestalgorithm-machine-learing/
- https://www.kdnuggets.com/2016/07/support-vectormachines-simple-explanation.html
- www.phishtank.com
- https://towardsdatascience.com/phishing-domain-detection-with-ml-5be9c99293e5