

# **WEB PHISHING DETECTION**

**TEAM ID:** PNT2022TMID00733

**BATCH NO:** B5-5M1E

**TECHNOLOGY:** APPLIED DATASCIENCE

## **TEAM MEMBERS:**

- 1. SARAVANAKUMAR R - 211419104239**
- 2. SRICHARAN S - 211419104265**
- 3. SRIRAM R - 211419104266**
- 4. SRI SANJAY S - 211419104268**

# **INDEX**

<b>CHAPTER NO</b>	<b>TITLE</b>	<b>PAGE.NO</b>
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 Project Overview	1
	1.2 Purpose	2
<b>2</b>	<b>LITERATURE SURVEY</b>	<b>2</b>
	2.1 Existing problem	2
	2.2 References	3
	2.3 Problem Statement Definition	4
<b>3</b>	<b>IDEATION &amp; PROPOSED SOLUTION</b>	<b>4</b>
	3.1 Empathy Map Canvas	4
	3.2 Ideation & Brainstorming	5
	3.3 Proposed Solution	6
	3.4 Problem Solution fit	8
<b>4</b>	<b>REQUIREMENT ANALYSIS</b>	<b>10</b>
	4.1 Functional requirement	10
	4.2 Non-Functional requirements	11
<b>5</b>	<b>PROJECT DESIGN</b>	<b>12</b>
	5.1 Data Flow Diagrams	12
	5.2 Solution & Technical Architecture	13
	5.3 User Stories	14

<b>6</b>	<b>PROJECT PLANNING &amp; SCHEDULING</b>	<b>16</b>
	<b>6.1</b> Sprint Planning & Estimation	16
	<b>6.2</b> Sprint Delivery Schedule	16
	<b>6.3</b> Reports from JIRA	17
<b>7</b>	<b>CODING &amp; SOLUTIONING</b>	<b>18</b>
	<b>7.1</b> Feature 1	18
	<b>7.2</b> Feature 2	18
	<b>7.3</b> Database Schema (if Applicable)	19
<b>8</b>	<b>TESTING</b>	<b>21</b>
	<b>8.1</b> Test Cases	21
	<b>8.2</b> User Acceptance Testing	21
<b>9</b>	<b>RESULTS</b>	<b>24</b>
	<b>9.1</b> Performance Metrics	24
<b>10</b>	<b>ADVANTAGES &amp; DISADVANTAGES</b>	<b>25</b>
	<b>10.1</b> Advantages	25
	<b>10.2</b> Disadvantages	25
<b>11</b>	<b>CONCLUSION</b>	<b>25</b>
<b>12</b>	<b>FUTURE SCOPE</b>	<b>26</b>
<b>13</b>	<b>APPENDIX</b>	<b>26</b>
	<b>13.1</b> Source Code	26
	<b>13.2</b> GitHub & Project Demo Link	38

# **1. INTRODUCTION:**

## **1.1. Project Overview:**

Phishing costs Internet users billions of dollars per year. It refers to luring techniques used by identity thieves to fish for personal information in a pond of unsuspecting Internet users. Phishers use spoofed e-mail, phishing software to steal personal information and financial account details such as usernames and passwords. This paper deals with methods for detecting phishing Web sites by analyzing various features of benign and phishing URLs by Machine learning techniques. We discuss the methods used for detection of phishing Web sites based on lexical features, host properties and page importance properties. We consider various machine learning algorithms for evaluation of the features in order to get a better understanding of the structure of URLs that spread phishing. The fine-tuned parameters are useful in selecting the apt machine learning algorithm for separating the phishing sites from benign sites. The criminals, who want to obtain sensitive data, first create unauthorized replicas of a real website and e-mail, usually from a financial institution or another company that deals with financial information. The e-mail will be created using logos and slogans of a legitimate company. The nature of website creation is one of the reasons that the Internet has grown so rapidly as a communication medium, it also permits the abuse of trademarks, trade names, and other corporate identifiers upon which consumers have come to rely as mechanisms for authentication. Phisher then send the "spoofed" e-mails to as many people as possible in an attempt to lure them in to the scheme. When these e-mails are opened or when a link in the mail is clicked, the consumers are redirected to a spoofed website, appearing to be from the legitimate entity.

## **1.2. Purpose**

The main purpose of the project is to detect the fake or phishing websites who are trying to get access to the sensitive data or by creating the fake websites and trying to get access of the user personal credentials. We are using machine learning algorithms to safeguard the sensitive data and to detect the phishing websites who are trying to gain access on sensitive data.

## **2. LITERATURE SURVEY**

### **2.1. Existing problem**

The purpose or goal behind phishing is data, money or personal information stealing through the fake website. The best strategy for avoiding the contact with the phishing web site is to detect real time malicious URL. Phishing websites can be determined on the basis of their domains. They usually are related to URL which needs to be registered (low-level domain and upper-level domain, path, query). Recently acquired status of intra-URL relationship is used to evaluate it using distinctive properties extracted from words that compose a URL based on query data from various search engines such as Google and Yahoo. These properties are further led to the machine-learning based classification for the identification of phishing URLs from a real dataset. This paper focus on real time URL phishing against phishing content by using phish-STORM. For this a few relationships between the register domain rest of the URL are consider also intra URL relentless is consider which help to dusting wish between phishing or non-phishing URL. For detecting a phishing website certain typical blacklisted URLs are used, but this technique is unproductive as the duration of phishing websites is very short. Phishing is the name of avenue. It can be defined as the manner of deception of an organization's customer to communicate with their confidential

information in an unacceptable behaviour. It can also be defined as intentionally using harsh weapons such as Spasm to automatically target the victims and targeting their private information. As many of the failures being occurred in the SMTP are exploiting vectors for the phishing websites, there is a greater availability of communication for malicious message deliveries. Proposed a novel classification approach that use heuristic-based feature extraction approach. In this, they have classified extracted features into different categories such as URL Obfuscation features, Hyperlink-based features. Moreover, proposed technique gives 92.5% accuracy. Also, this model is purely depending on the quality and quantity of the training set and Broken links feature extraction.

## **2.2. References:**

[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-attackstatistics/#gref>

[3] Mahmoud Khonji, Youssef Iraqi, "Phishing Detection: A Literature Survey IEEE, and Andrew Jones, 2013.

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

[5] <http://dataaspirant.com/2017/01/30/how-decision-treealgorithm-works/>

[6] <http://dataaspirant.com/2017/05/22/random-forestalgorithm-machine-learning/>

[7] <https://www.kdnuggets.com/2016/07/support-vectormachines-simple-explanation.html>

[8] [www.alexa.com](http://www.alexa.com)

[9] [www.phishtank.com](http://www.phishtank.com)

### 2.3. Problem Statement Definition:

Phishing is one of the techniques which are used by the intruders to get access to the user credentials or to gain access to the sensitive data. This type of accessing the is done by creating the replica of the websites which looks same as the original websites which we use on our daily basis but when a user clicks on the link, he will see the website and think its original and try to provide his credentials. To overcome this problem, we are using some of the machine learning algorithms in which it will help us to identify the phishing websites based on the features present in the algorithm. By using these algorithms, we can be able to keep the user personal credentials or the sensitive data safe from the intruders.

## 3. IDEATION & PROPOSED SOLUTION


### 3.1. Empathy Map Canvas



## 3.2. Ideation & Brainstorming




### Step-1: Team Gathering, Collaboration and Select the Problem Statement


Template



## Web Phishing Detection


Web Phishing is a form of activity 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.

 10 minutes to prepare  
 1 hour to collaborate  
 2-8 people recommended



**Before you collaborate**

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

 10 minutes

A

**Team gathering**

Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B


**Set the goal**

Think about the problem you'll be focusing on solving in the brainstorming session.

C

**Learn how to use the facilitation tools**


Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) 

1


**Define your problem statement**

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.


 5 minutes


**PROBLEM**


How to determine the best set of features to be used for phishing detection?  
How do I select the best algorithm out of many?  
How to enhance the performance of the algorithm that is has selected?  
How to overcome the false negative problem?


**Key rules of brainstorming**


To run an smooth and productive session


 Stay in topic.

 Encourage wild ideas.

 Defer judgment.

 Listen to others.

 Go for volume.

 If possible, be visual.



## Step-2: Brainstorm, Idea Listing and Grouping

2

### Brainstorm

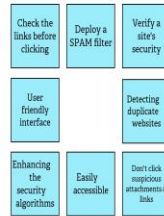
Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

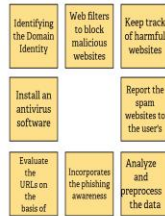
#### Saravanakumar R



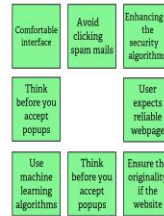
#### Sricharan S



#### Sriram R



#### Sri Sanjay S



3

### Group ideas

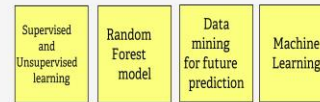
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

🕒 20 minutes

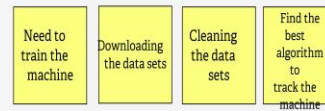
#### Category 1: Technologies used



#### Category 2: Detection Algorithm



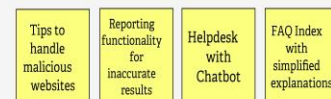
#### Category 3: Training the machine



#### Category 4: Security



#### Category 5: Additional Features



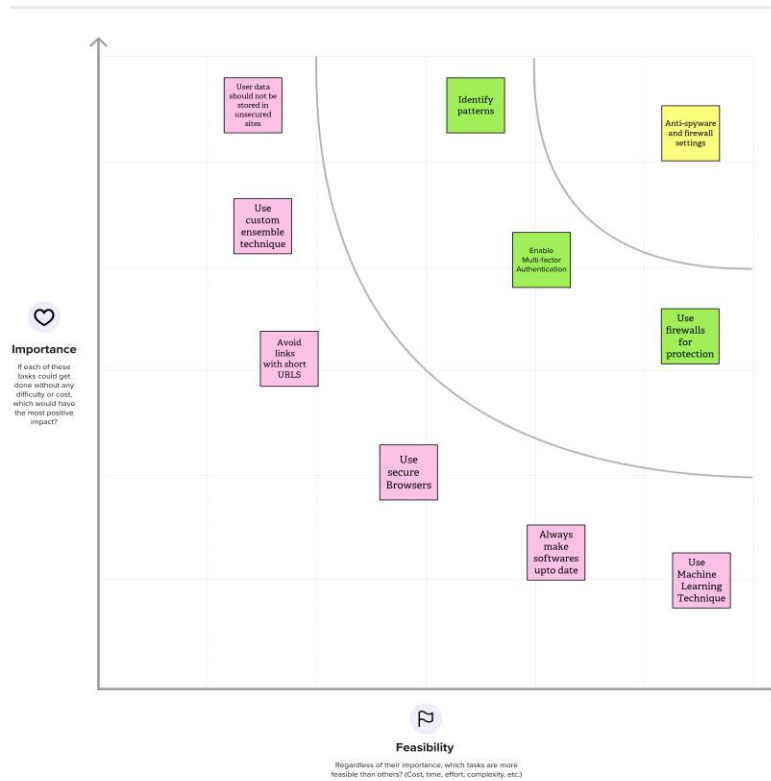
## Step-3: Idea Prioritization

4

### Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes



5

### After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

#### Quick add-ons

- A Share the mural**  
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.
- B Export the mural**  
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

#### Keep moving forward

- Strategy blueprint**  
Define the components of a new idea or strategy.  
[Open the template →](#)
- Customer experience journey map**  
Understand customer needs, motivations, and obstacles for an experience.  
[Open the template →](#)
- Strengths, weaknesses, opportunities & threats**  
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.  
[Open the template →](#)

[Share template feedback](#)

## 3.3. Proposed Solution

The purpose or goal behind phishing is data, money or personal information stealing through the fake website. The best strategy for avoiding the contact with the phishing web site is to detect real time malicious URL. Phishing websites can be determined on the basis of their domains. They usually are related to URL which needs to be registered (low-level domain and upper-level domain, path, query). Recently acquired status of intro-URL relationship is used to evaluate it using distinctive properties extracted from words that compose a URL based on query data from various search engines such as Google and Yahoo. These

properties are further led to the machine-learning based classification for the identification of phishing URLs from a real dataset. This paper focus on real time URL phishing against phishing content by using phish-STORM. For this a few relationships between the register domain rest of the URL are consider also intro URL relentless is consider which help to dusting wish between phishing or non-phishing URL. For detecting a phishing website certain typical blacklisted URLs are used, but this technique is unproductive as the duration of phishing websites is very short. Phishing is the name of avenue. It can be defined as the manner of deception of an organization's customer to communicate with their confidential information in an unacceptable behaviour. It can also be defined as intentionally using harsh weapons such as Spasm to automatically target the victims and targeting their private information. As many of the failures being occurred in the SMTP are exploiting vectors for the phishing websites, there is a greater availability of communication for malicious message deliveries. Proposed a novel classification approach that use heuristic-based feature extraction approach. In this, they have classified extracted features into different categories such as URL Obfuscation features, Hyperlink-based features. Moreover, proposed technique gives 92.5% accuracy. Also, this model is purely depends on the quality and quantity of the training set and Broken links feature extraction.

### 3.4. Problem Solution fit

Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> <b>CS</b> Who is your customer? <ul style="list-style-type: none"> <li>✓ The customer focus is on people who use the internet for e-transactions where safeguarding customers data important and vital.</li> <li>✓ Government agencies and industries are another customer base where they require phishing detection systems to safeguard confidential information.</li> </ul>	<b>6. CUSTOMER CONSTRAINTS</b> <b>CC</b> What constraints prevent your customers from taking action or limit their choices of solutions? <ul style="list-style-type: none"> <li>✓ Lack of basic knowledge in verifying the correct URL of the webpage.</li> <li>✓ Lack of user testing by organization as they require more resources and money. They are always in a rush which makes them prone to errors.</li> <li>✓ Malwares have become more complex than what a layman can understand.</li> </ul>	<b>5. AVAILABLE SOLUTIONS</b> <b>AS</b> 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? <ul style="list-style-type: none"> <li>✓ Using a good Antivirus software or an Anti-Phishing toolbar which are available as extensions in browsers.</li> <li>✓ Verifying the websites privacy policy and ensuring the websites are SSL certified.</li> <li>✓ Double checking the domain name.</li> <li>✓ Anti-Spam software and Blacklisting.</li> </ul>	Explore AS, differentiate
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <b>J&amp;P</b> Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides. <ul style="list-style-type: none"> <li>✓ The phishing websites must be detected prior and should be blacklisted.</li> <li>✓ Building a phishing URL detecting website where the user can copy paste the URL and find if the URL is legitimate.</li> <li>✓ Companies trust is broken if private data of customers are leaked.</li> </ul>	<b>9. PROBLEM ROOT CAUSE</b> <b>RC</b> What is the real reason that this problem exists? What is the back story behind the need to do this job? <ul style="list-style-type: none"> <li>✓ Lack of basic awareness among the common folk and leniency in the adaption of new security measures.</li> <li>✓ Low-cost phishing and ransomware tools are easy to get hold of.</li> <li>✓ The financial incentive is high which makes more people to launch phishing attacks despite of the consequences.</li> </ul>	<b>7. BEHAVIOUR</b> <b>BE</b> What does your customer do to address the problem and get the job done? <ul style="list-style-type: none"> <li>✓ Customers should take a "trust no one" approach when opening an email and should always verify the "From" address of the email.</li> <li>✓ Avoid clicking links or attachments in emails from unfamiliar sources and change your passwords regularly.</li> </ul>	Focus on J&P, tap into BE, understand RC
Identify strong TR & EM	<b>3. TRIGGER</b> <b>TR</b> What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news. <ul style="list-style-type: none"> <li>✓ To prevent data including login credentials and credit card numbers from getting stolen.</li> <li>✓ Seeing others lose Money due to phishing and their reputation getting damaged. This increases the awareness of the people.</li> </ul>	<b>10. YOUR SOLUTION</b> <b>SL</b> 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. This project trains a model that can automatically classify the phishing websites at run-time without any human interventions. We would create an interactive webpage using HTML & Python where a user may enter an URL and the system classifies whether it is a phishing or a legitimate webpage using Machine learning (ML) algorithms and then provide the user with appropriate feedback. If it is a phished webpage, then it will be added to the blacklist. This provides better data confidentiality to the users.	<b>8. CHANNELS OF BEHAVIOUR</b> <b>CH</b> <b>8.1 ONLINE</b> What kind of actions do customers take online? <ul style="list-style-type: none"> <li>✓ By using appropriate firewalls and not clicking random pop ups in browsers and in email links.</li> <li>✓ Using a secure Wi-Fi network for online transactions and always double checking the URL twice beforehand.</li> </ul>	Extract online & offline CH of BE
	<b>4. EMOTIONS: BEFORE / AFTER</b> <b>EM</b> How do customers feel when they face a problem or a job and afterwards? <b>BEFORE:</b> 1. They feel threatened and insecure using the internet. 2. Fear of insecurity, Subject to vulnerability. <b>AFTER:</b> 1. Data confidentiality. 2. Secure Transactions.		<b>8.2 OFFLINE</b> What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development. <ul style="list-style-type: none"> <li>✓ Not sharing confidential information in spam phone calls or in random messages.</li> <li>✓ Raising awareness by conducting small camps in your locality among the elderly and people who have less computer knowledge.</li> </ul>	

## 4. REQUIREMENT ANALYSIS

### 4.1. Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Register by entering details such as name, email, password.
FR-2	User Confirmation	Login using the registered email id and Password.
FR-3	Website Comparison	Blacklist filtering and Whitelist filtering techniques are used to compare the website URL.
FR-4	Feature Selection	Based on the length of an URL, number of dots in URL and check for the correct spelling and grammar.
FR-5	Feature Vectorization	Training and Testing datasets should be developed.
FR-6	Classifier	Model sends all output 10 classifier and produces final result.
FR-7	Results	Model then displays whether website is a legitimate or a phishing site.

## 4.2. Non-Functional requirements

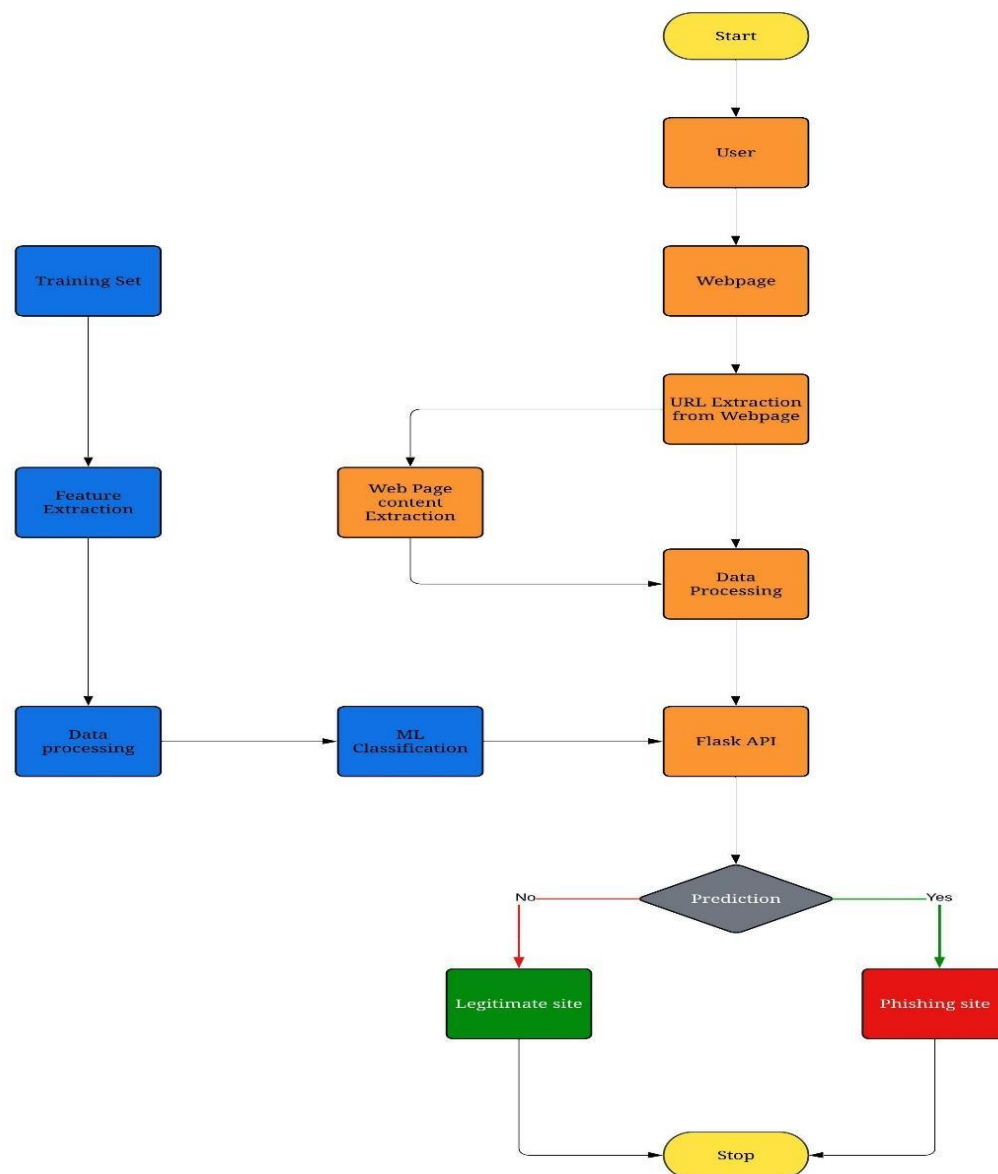
Following are the non-functional requirements of the proposed solution.

<b>NFR No.</b>	<b>Non-Functional Requirement</b>	<b>Description</b>
NFR-1	<b>Usability</b>	User can access to several website easily using web phishing detection without losing any data.
NFR-2	<b>Security</b>	Alert message must be sent to the users to enable secure browsing.
NFR-3	<b>Reliability</b>	The web phishing websites must detect accurately and the result must be reliable.
NFR-4	<b>Performance</b>	The performance should be faster and user friendly for the effective performance.
NFR-5	<b>Availability</b>	The system will be accessible to the user at any point in time through a web browser.
NFR-6	<b>Scalability</b>	It must be able to handle an increase in users and loads without disrupting the end users.

## 5. PROJECT DESIGN

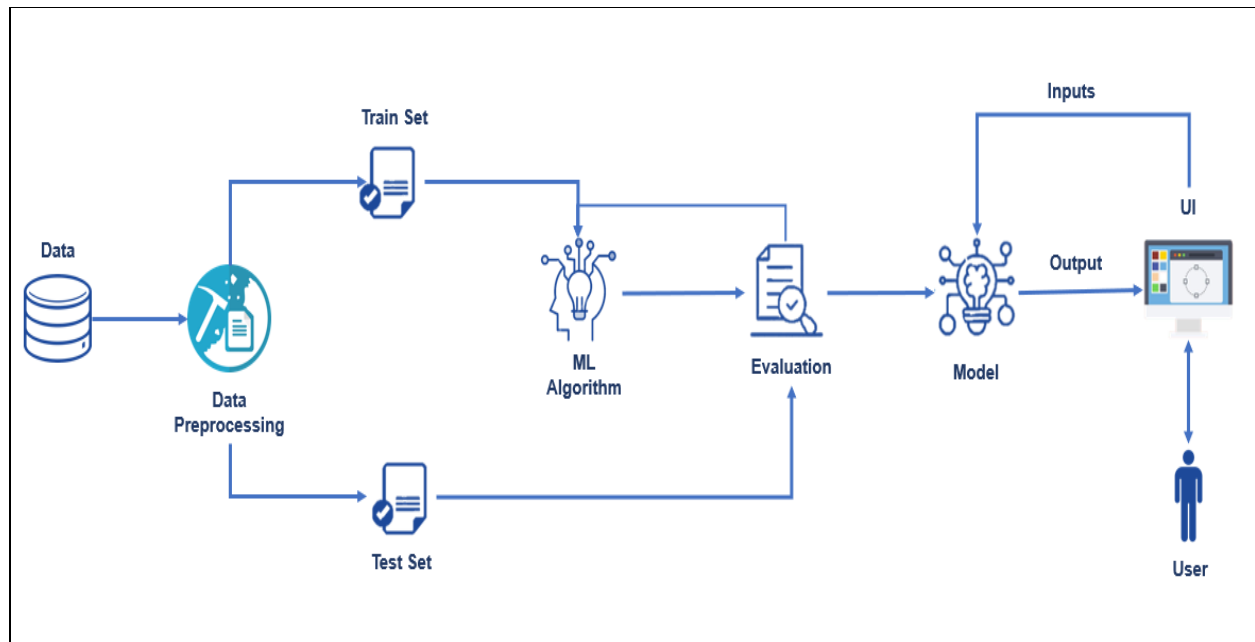
### 5.1. Data Flow Diagrams

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



**DATAFLOW DIAGRAM**

## 5.2. Solution & Technical Architecture





### 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 application by entering my email, password, and confirming my password.	I can access my account / dashboard.	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application.	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook.	I can register & access the dashboard with Facebook Login.	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail.		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password.		High	Sprint-1
	Dashboard					Sprint-1
Customer (Web user)	User input	USN-1	As a user I can input the particular URL in the required field and waiting for validation.	I can go access the website without any problem	High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story/ Task	Acceptance criteria	Priority	Release
Customer Care Executive	Feature extraction	USN-1	After I compare in case if none found on comparison then we can extract feature using heuristic and visual similarity approach.	I can have comparison between websites for security.	High	Sprint-1
Administrator	Prediction	USN-1	Here the Model will predict the URL websites using Machine Learning algorithms such as Logistic Regression, and KNN to forecast the URL of the websites.	I can accurately forecast the specific algorithms in this way.	High	Sprint-1
	Classifier	USN-2	To create the final product, I will now feed all the model output to the classifier.	I'll use this to identify the appropriate classifier for generating the outcome.	Medium	Sprint-2

## 6. PROJECT PLANNING & SCHEDULING

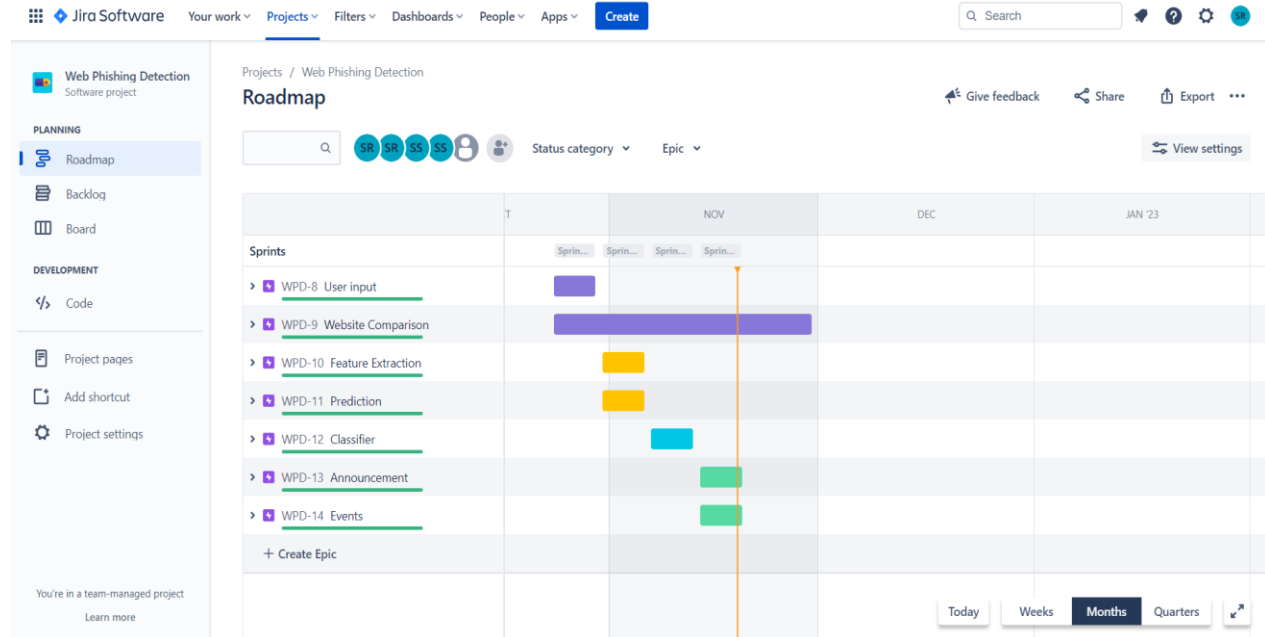
### 6.1. Sprint Planning & Estimation

Setting Up Application Environment To create lots of environment. Create or Enrolment to the IBM cloud, Docker CLI installation, Implementing Web Application. We create a UI to interact with application. Create database system DB2 and connect it with python and integrate with Phishing Model API. Integration with python code for include some RestAPI services for to give a Phished URLs. Deployment of App in IBM Cloud In the deploy process, the deployment in Kubernetes cluster is the major task before that we need to containerize the app and upload image to IBM container Registry.

### 6.2. Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	16 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

## 6.3. Reports from JIRA



## **7. CODING & SOLUTIONING**

### **7.1. Feature 1**

**Instant Information:** The phrase “just Google it” holds special relevance in our day to day lives. It is the mobile apps that allow you to search the content instantly anywhere, anytime. Apps are reported to be 1.5 times faster as compared to websites and perform actions at a faster pace too. Human minds are inquisitive and smartphone applications serve as an instant relief for this inquisitiveness. **Instant Connection & Communication:** Mobile phones and the internet have made this world a small place. There are ample of apps that allow a person to instantly connect and communicate another person living miles away. Apps have played a vital role in helping us in maintaining a healthy work-life balance. **Better Productivity:** According to a report released by Accenture, business employees acknowledge the fact that smartphone apps play a critical role in their smooth business operations. Earlier people can access professional emails and messages from office computers only. Now with the help of various apps and an internet connection they can check their emails and messages from anywhere in the world. Even if you’re holidaying with your family, you can stay connected with your business updates with just a simple tap. This will contribute to your productivity as well.

### **7.2. Feature 2**

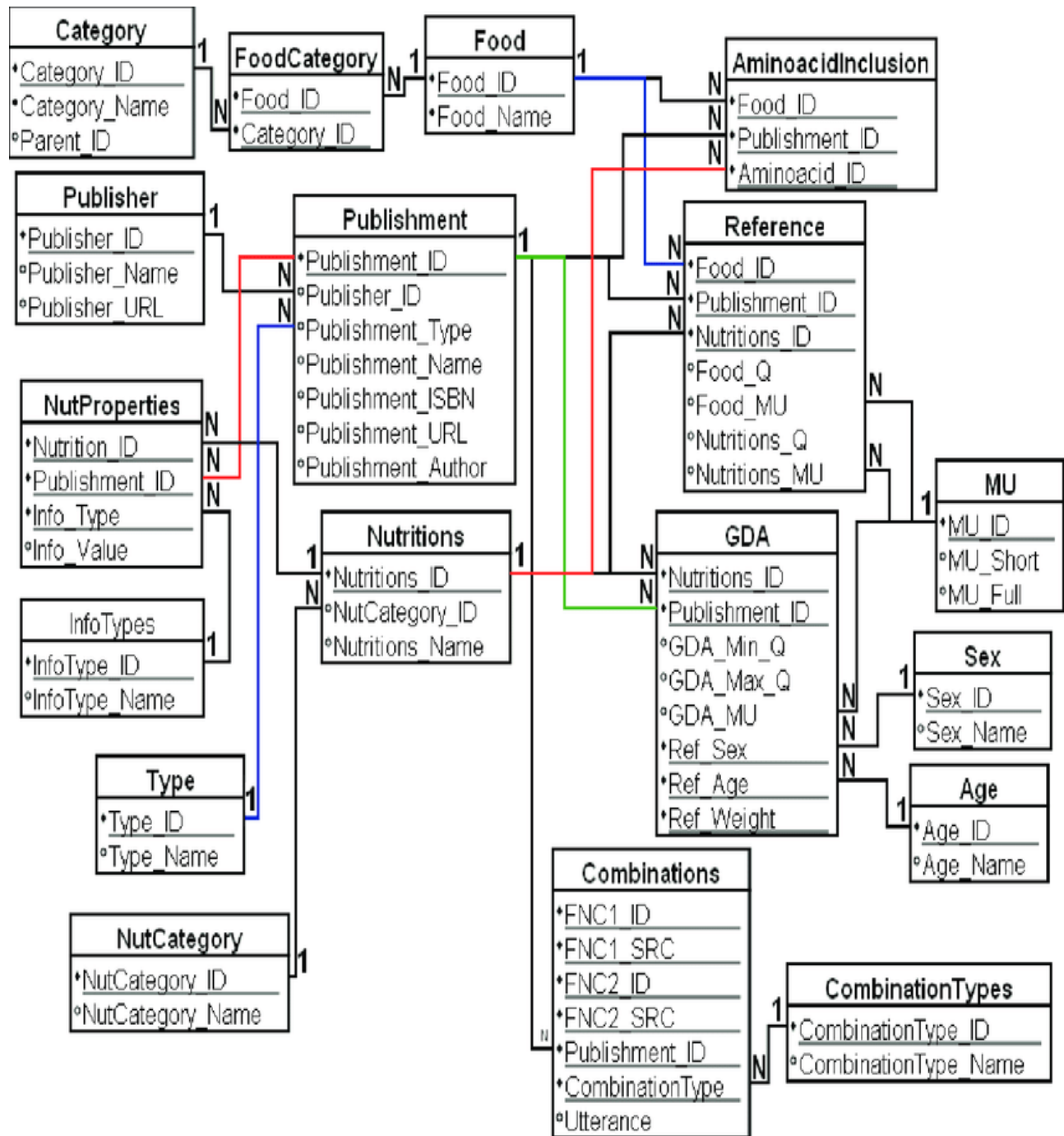
**Personal Accounts and Registration:** Personalization is the key feature of every health and nutrition app. If a user can feed his personal health details and can get suitable advice, what more they can ask for? Make sure you have a feature to gather and analyze every user’s data pertaining to his or her weight, age, eating habits, allergies, level of physical activity, weight goals, and so on. **Food Logging and Dashboard:** A top-notch health and nutrition app should have a duly designed

dashboard and a food logging feature. You need to prepare a database consisting of all diet information in it like calories, food intake, energy proportion, hydration meter and so on. It is almost impossible to develop this database on your own so why not use a ready-made API for this purpose? You can get this application programming interface for free. All you need to do is fill a form on the respective company's website and wait for the approval. Simple isn't it

**Push Notifications:**

Push notifications hold a special relevance in smartphone apps. If you too wish to go for a healthcare mobile app development, make sure you don't lag behind in this! Install this feature to remind your users for their upcoming meal and motivate them to achieve their health and nutrition goals. Control the frequency of notifications as it can easily annoy the users.

### 7.3. Database Schema



## **8. TESTING**

### **8.1. Test Cases**

With a concerted effort, I conducted research on general well-being to have a rudimentary grasp on health management, as well as the existing nutrient tracking apps in order to get an understanding of what is already existing in the market, the characteristics, specialties, and usability. There are a considerable number of nutrition tracking apps existing in the market. They aim to track daily calories/macros intake by logging meals to achieve users' preset goals. To log meals, users can input the food in the app, or scan the barcode of a package. Most apps allow users to connect with associated activities apps to track exercise progress. With a premium upgrade, users can get access to tailor-made recipes according to health goals or specified diets. In order to build a realistic initial target group, I wanted to conduct some usability tests with 5 users that regularly engage in physical activity and food tracking, including both first-time and regular users of meal planning and fitness apps. I asked these individuals to perform tasks related to general usage of the MyFitnessPal, Lifesum, and Nutrition Coach apps (such as food logging, searching, and checking their caloric breakdown.)

### **8.2. User Acceptance Testing**

Must-have features of a diet and nutrition app I wanted to address the user pain points by including (and improving) the core features of the application. Personal profiles After downloading the app, a user needs to register and create an account. At this stage, users should fill in personal information like name, gender, age, height, weight, food preferences, allergies, and level of physical activity. Food logging and dashboard Allowing users to analyze their eating habits. They should be able to log food and water intake and see their progress on a dashboard that can

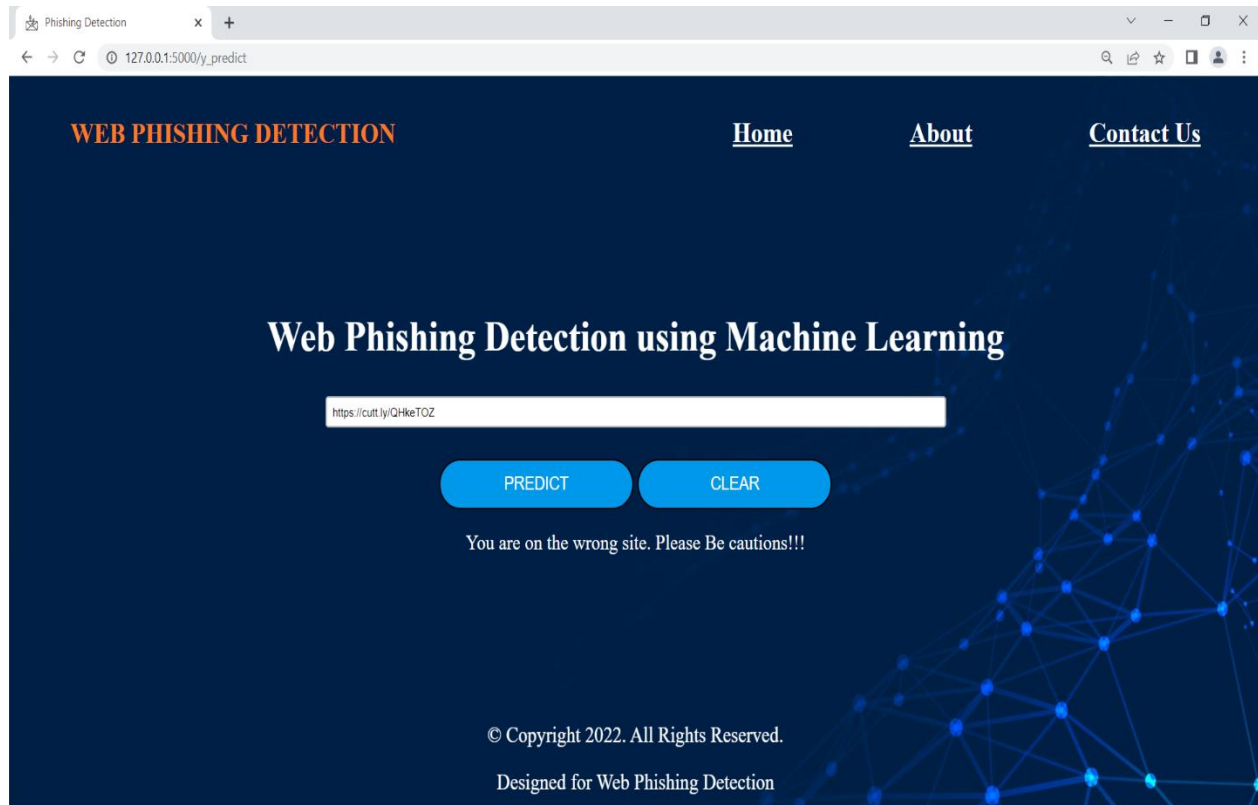
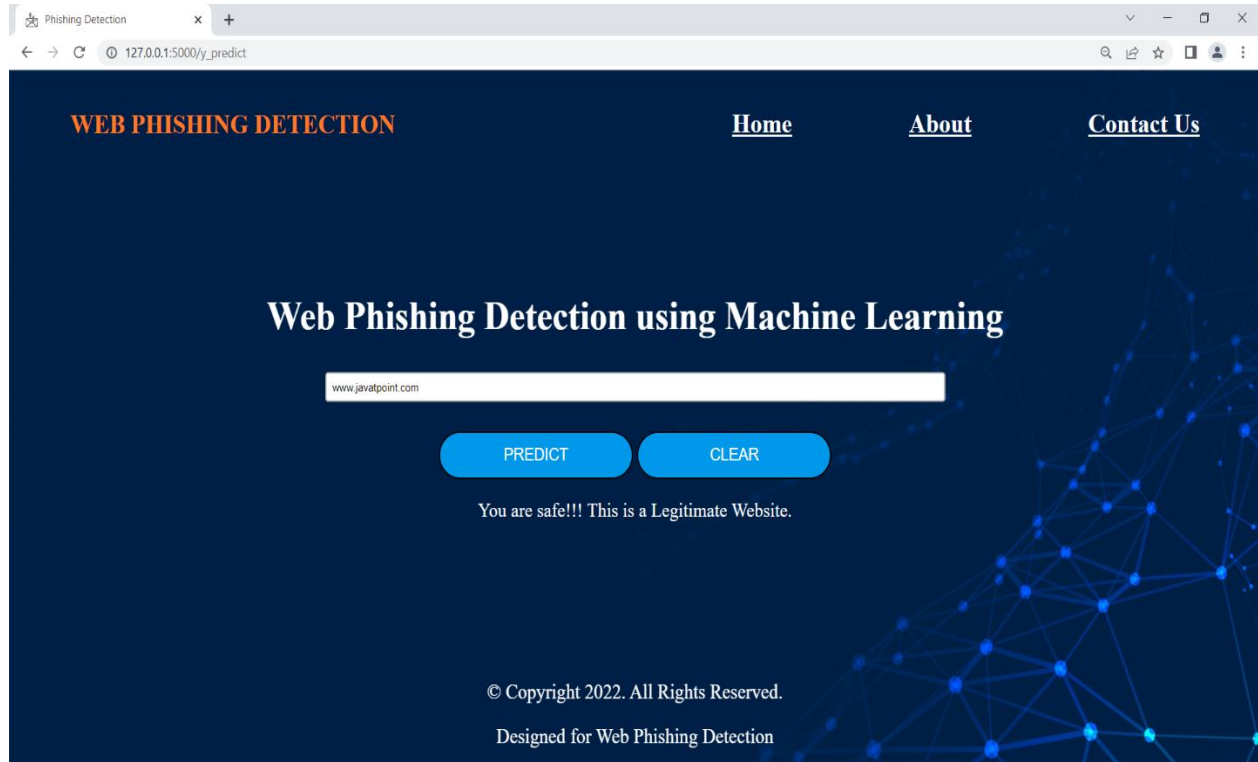


track calories, fat, protein, and carbs. Push notifications Push notifications are an effective tool for increasing user engagement and retention. To motivate users to keep moving toward their goals, it's pertinent to deliver information on their progress toward the current goal and remind them to log what they eat. Calorie counter Enabling the application to calculate calories users have burned and eaten based on the data they've logged. Barcode scanner Let users count calories and see accurate nutrition information via a built-in barcode scanner. Recipe book Users will appreciate the opportunity to find healthy recipes in the app. Including pictures, videos, and even voice instructions in your recipes would be a valuable feature. Also, allowing users to rate recipes and sort them by keywords, ingredients, categories, and calories. Diet plans Help users maintain a healthy diet by offering diet plans. Usually, a diet plan includes meal suggestions, nutritional tips, recipes, and recommended total calorie intake per week or day. Gamification Including game elements to increase user engagement and retention. Using ranks, badges, and points to reward users for achievements such as losing weight or completing goals. Integration with wearables. There are different trackers and wearables to integrate with. For example, Apple Watch, Android Wear, Jawbone, Fitbit, and Samsung Gear to synchronize data on physical activity and health metrics. Nice-to-have features of a diet and nutrition app Since nutrition apps can have different purposes, their functionality can differ accordingly. Below are features that I considered including later on or that could be useful for some nutrition apps. Blog A lot of users want a diet and nutrition application to not only count calories but also share some diet tips to help them improve eating habits. This is where blogs come in handy. There, the latest food and nutrition research, news, and health tips could be shared. Shopping list Letting users import ingredients from a diet plan or a recipe to a shopping list or add groceries manually. Experts Users will definitely appreciate being able to get in touch with diet coaches for expert advice. This could

be a paid feature. User Personas From the interview and observation sessions, I collected and synthesized some information that can be used in the upcoming design process. I could clearly define 2 user profiles from the research data. A casual dieter who does not follow a health plan regularly: Enise is a full-time student who needs reminders, suggestions, and coaching to cook more often with fresh ingredients because they want to stay on top of their health and make it a part of their routine.

## 9. RESULTS

### 9.1. Performance Metrics



## **10. ADVANTAGES & DISADVANTAGES**

### **10.1 Advantages**

1. This system can be used by many E-commerce or other websites in order to have good customer relationship.
2. User can make online payment securely.
3. Data mining algorithm used in this system provides better performance as compared to other traditional classifications algorithms.
4. With the help of this system user can also purchase products online without any hesitation.
5. Output of the screen is easy understandable.

### **10.2 Disadvantages**

1. If Internet connection fails, this system won't work.
2. All websites related data will be stored in one place.

## **11. CONCLUSION**

Certain classifiers that are more prone to overfitting than others are present, thus yielding higher accuracy rate if they are based on the dataset that they have been trained upon. This result can be observed in the experiment performed through Azure, specifically trees. To address the overfitting problem while focusing on increasing the prediction accuracy, the proposed solution model uses feature selection and ensemble learning where multiple learning models are combined to produce a prediction. By using multiple models, the prediction is not bias towards one model and is instead based on majority of predictions such that all predictions from each model influences the final ensemble prediction.

## **12. FUTURE SCOPE**

Phishing is a growing problem for internet users. There are a number of anti-phishing tools available to cope against this problem. Still there are limitation on accuracy because detection techniques are time consuming. Among several machine learning algorithm, Random- forest gives the better result. This work become unique from other existing work by proposing a group of features that can be extracted automatically using our own software tool. In future we can make the system available in mobile devices.

## **13. APPENDIX**

### **13.1. Source Code**

#### **app.py**

```
import numpy as np
import pickle
import inputScript
from flask import Flask, render_template, request
app = Flask(__name__)

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

@app.route("/")
def index():
    return render_template("index.html")

@app.route("/about")
def about():
    return render_template("about.html")
```

```
@app.route("/contact")
def contact():
    return render_template("contact.html")

@app.route("/predict")
def predict():
    return render_template("final.html")

@app.route("/y_predict", methods = ['GET', 'POST'])
def y_predict():
    geturl = request.form['url']
    check_prediction = inputScript.main(geturl)
    prediction = model.predict(check_prediction)
    print(prediction)
    output = prediction[0]
    if(output==1):
        pred ='You are safe!!! This is a Legitimate Website.'
    else:
        pred = 'You are on the wrong site. Please Be cautions!!!'
    return render_template('final.html',url_path = geturl,url = pred)

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

## index.html

```
<html>

  <head>

    <title>Web Phishing Detection</title>

    <link rel="icon" type="image/x-icon" href="static/index.png">

    <link rel="stylesheet" href="{ { url_for('static', filename = 'index.css')
  }}">


  </head>

  <body>

    <div id="blue">

      <h1 id="title">WEB PHISHING DETECTION<span style="float:right"><a
href="">Home</a> <a href="">about</a> <a href="">contact</a> <a href="">Contact
Us</a><a href="">predict</a></span></h1>

      <br>

      </img>

      <h1 id="sub-t">Best Solution to Detect Phishing Websites</h1> <br>

      <p>Be aware of what's happening with you confidential data</p>

      <span> <button style="font-size: 12px" id="start"><a
href="">predict</a></button> </span></div>

    <div id="white">

      <footer>

        <div class="copyright">

          <div class="container">

            <div class="row">
```

```
<div class="col-md-10 offset-md-1">
    <center><h2>&copy; Copyright 2022. All Rights
Reserved.</h2></center>
    <center><h2><center>Designed for Web Phishing
Detection</center></h2></center>
</div>
</div>
</div>
</div>
</footer>
</body>
</html>
```

### **about.html**

```
<!DOCTYPE html>
<html>
<head>
    <title>About Us</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet"
href="https://pro.fontawesome.com/releases/v5.10.0/css/all.css">
    <link rel="icon" type="image/x-icon" href="static/about.png">
    <link rel="stylesheet" href="{ { url_for('static', filename = 'about.css') } }">
</head>
<body>
    <div id="blue">
```



```
<h1 id="title">WEB PHISHING DETECTION<span
style="float:right"><a href="/">Home</a> <a href="/contact">Contact
Us</a><a href="/predict">Get Started</a></span></h1>
```

```
<div class="section">
```

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

```
<div class="content-section">
```

```
<div class="title">
```

```
<h1>About Us</h1>
```

```
</div>
```

```
<div class="content">
```

```
<h3>It is very important to take care of the users
personal sensitive data, that is being stolen by the phishers. </h3>
```

```
<p>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. In order to predict the phishing websites, we
proposed an intelligent, flexible and effective system that uses Machine
Learning Algorithm inorder to extract the phishing datasets criteria to classify
their legitimacy. This system provides better performance as compared to other
traditional Detection algorithms. With the help of this system the user can
easily able to understand whether the wesite is phishing or a legitimate
one.</p>
```

```
<br><form><br><br><br>
```

```
<label for="lname"><u><h2>CONTACT
US</h2></u></label>
```

```
</form>
```

```
</div>
```

```
<div class="social">
    <div class="social-icons">
        <a href="#"><i class="fab fa-
facebook"></i><i class="fab fa-twitter"></i><i class="fab fa-linkedin-
in"></i></a>

    </div>
</div>

<div class="image-section">
    
</div>
</div>
</div>
<div id="white">
    <footer>
        <div class="copyright">
            <div class="container">
                <div class="row">
                    <div class="col-md-10 offset-md-1">
                        <center><h2>&copy; Copyright 2022. All Rights
Reserved.</center></h2>
                        <center></p><h2><center>Designed for Web Phishing
Detection</center></h2></center></p>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

```
        </footer>
    </div>
</body>
</html>
```

### **final.html**

```
<html>
    <head>
        <title>Phishing Detection</title>
        <link rel="icon" type="image/x-icon" href="static/predict.png">
        <link rel="stylesheet" href="{ { url_for('static', filename = 'final.css') } }">
    </head>
    <body background="new.jpg">
        <div id="blue">
            <h1 id="title">WEB PHISHING DETECTION<span
style="float:right"><a href="\ ">Home</a> <a href="\about">About</a> <a
href="\contact">Contact Us</a></span></h1>
        </div>
        </div>
        <div id="white">
            <center>
                <form action="{ { url_for('y_predict') } }" method="post">
                <h1><b><center>Web Phishing Detection using Machine
Learning</center></b></h1>
                <input type="text" placeholder="Enter the URL to be searched"
id="url" name="url" value="{ { url_path } }" size="20"><br><br>
```

```
<input type="submit" style="font-size:20px" id="start"
value="PREDICT">
<input type="reset" style="font-size:20px" id="start"
value="CLEAR">
</form>
{{ url }}
</center>
</div>
<footer>
<div class="copyright">
<div class="container">
<div class="row">
<div class="col-md-10 offset-md-1">
<center>&copy; Copyright 2022. All Rights
Reserved.</center>
<center></p><center>Designed for Web Phishing
Detection</center></p>
</div>
</div>
</div>
</footer>
</body>
</html>
```

## contact.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>Contact Us</title>
  <link rel="stylesheet"
href="https://fonts.googleapis.com/css2?family=Roboto:wght@300;400;500;7
00&display=swap">
  <!-- Font Awesome CDN -->
  <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.4/css/all.css">
  <link rel="icon" type="image/x-icon" href="static/contact.png">
  <link rel="stylesheet" href="{ { url_for('static',filename = 'contact.css') } }">
</head>
<body>
  <div><h1 id="title">WEB PHISHING DETECTION<span
style="float:right"><a href="">Home</a> <a href="\about">About</a>
</a><a href="\predict">Get Started</a></span></h1>
  <section id="section-wrapper">
    <div class="box-wrapper">
      <div class="info-wrap">
        <h2 class="info-title">Contact Information</h2>
        <h3 class="info-sub-title">Fill up the form and our Team will get
back to you within 24 hours</h3>
```

```
<ul class="info-details">
  <li>
    <i class="fas fa-phone-alt"></i>
    <span>Phone:</span> <a href="tel: +91-7555616326">+91-
7555616326</a>
  </li>
  <li>
    <i class="fas fa-paper-plane"></i>
    <span>E-mail:</span> <a
href="mailto:saravanakumar18081@gmail.com">web@info.com</a>
  </li>
  <li>
    <i class="fas fa-globe"></i>
    <span>Website:</span> <a href="#">webinfosite.com</a>
  </li>
</ul>
<ul class="social-icons">
  <li><a href="#"><i class="fab fa-facebook"></i></a></li>
  <li><a href="#"><i class="fab fa-twitter"></i></a></li>
  <li><a href="#"><i class="fab fa-linkedin-in"></i></a></li>
</ul>
</div>
<div class="form-wrap">
  <form action="mailto:saravanakumar18081@gmail.com"
method="POST">
    <h2 class="form-title">Send us a message</h2>
    <div class="form-fields">
```

```
<div class="form-group">
  <input type="text" class="fname" placeholder="First Name">
</div>
<div class="form-group">
  <input type="text" class="lname" placeholder="Last Name">
</div>
<div class="form-group">
  <input type="email" class="email" placeholder="E-Mail
ID">
</div>
<div class="form-group">
  <input type="number" class="phone"
placeholder="Phone.No">
</div>
<div class="form-group">
  <textarea name="message" id="" placeholder="Write your
message"></textarea>
</div>
</div>
<input type="submit" value="Send Message" class="submit-
button">
</form>
</div>
</div>
</div>
</section>
<div class="container">
```

```

        <button type="submit" class="btn"
onclick="openPopup()">Submit</button>
        <div class="popup" id="popup">
            
            <h2>Thank You!!!</h2>
            <p>Your details has been submitted successfully. We'll get back to you
soon...</p>
            <button type="button" onclick="closePopup()">OK</button>
        </div>
    </div>
    <script>
        let popup = document.getElementById("popup");
        function openPopup(){
            popup.classList.add("open-popup");
        }
        function closePopup(){
            popup.classList.remove("open-popup");

        }
    </script>
    <footer>
        <div class="copyright">
            <div class="container">
                <div class="row">
                    <div class="col-md-10 offset-md-1">
                        <br><br><br><br><br><br><br><center><h3>&copy;
Copyright 2022. All Rights Reserved.</h3></center>

```



```

        <center><h3><center>Designed for Web Phishing
Detection</center></h3></center>
    </div>
</div>
</div>
</div>
</footer>
</body>
</html>

```

## 13.2 GitHub & Project Demo Link

**GITHUB:** <https://github.com/IBM-EPBL/IBM-Project-4244-1658725622>

**PROJECT DEMO LINK:**

<https://drive.google.com/file/d/1vuAjA8CMhAHXdrkA8LZeePyuJmAQC0Eo/view?usp=sharing>