Project Report - Web Phishing Detection

Team ID: PNT2022TMID35524

Team Leader: Abhinand G (2019115003)

Team Member 1: Aruna Srikamakshi R (2019115020)

Team Member 2: Roshni Balasubramanian (2019115083)

Team Member 3: Suba Varshini V (20191150107)

1. INTRODUCTION

1.1 Project Overview

Security has become increasingly important due to the prevalence of hackers and cyber-thieves gaining access to our sensitive data and information. The project 'Web Phishing Detection' aims to reduce the potential to fall for web phishing and scams. The project uses machine learning algorithms and features dimensionality reduction with the use of a combined strategy that gains the best results. This model has been used in a simple UI such that any user can easily input the URL that they would like to visit to get a prediction on whether the site is safe.

1.2 Purpose

This project reduces the fear among common internet users of all age groups. In a 2021 research by IBM, it was confirmed that there was a two percent increase in web phishing attacks from 2019 to 2020. Around 86% of all organizations have experienced phishing even if that includes just a single employee. Data reveals that around 90% of all data breaches are due to phishing. In order to reduce the effects of this largely existing problem, rather than blindly opening a link without knowing its safety level, using our tool will give the user a ballpark around its safety. Users can safely browse the internet, without fearing of falling for scams or getting scammed.

2. LITERATURE SURVEY

2.1 Existing problem

The existing problem that can be observed in work is the limitation of exploring a wide variety of existing features that are not restricted to the domain. The majority work on web phishing also does not use feature selection or feature dimension reduction strategies to enhance the results of metrics like accuracy.

2.2 References

In Al-Sarem et Al's "An Optimized Stacking Ensemble Model for Phishing Websites Detection" paper, the work involves using genetic algorithms to tune parameters of several ensemble machine learning methods. These methods included Random Forest, AdaBoost, XGBoost and GradientBoost. They observed higher accuracies than the currently proposed models and reached a result of 97.39% for the dataset used. The limitation existing in their work is the lack of accounting for weightage of features. In M Sanchez-Paniagua et Al.'s "Phishing Websites

Detection Using a Novel Multipurpose Dataset and Web Technologies Features” explores a public dataset of PILWD-134K and classification using LightGBM. 54 Features were extracted which obtained a 97.95% accuracy. However, this work mentions that due to the significant number of samples in dataset there are false positive and negatives possible. They also mention that the verification process of the samples was directly dependent entirely on the PhishTank service, a preexisting phishing detection service. Several of the new hybrid features introduced by the work such as copyright, title and domain name may exclude websites without trademarks which could classify these sites as malicious. This is another major limitation to the proposed work.

In Mehmet Korkmaz et Al’s “Detection of Phishing Websites by Using Machine Learning-Based URL Analysis”, there has been utilization of 48 characteristics across 3 distinct dataset, along with the usage of 8 different methodologies. They have observed that on each of the three datasets, Random Forest Classifier has the highest accuracy, while another desired method of classification is using an Artificial neural network. For the three datasets, the maximum accuracies that have been achieved are 94.59%, 90.5% and 91% respectively. However, the limitation lies on the fact that the accuracy obtained hasn’t crossed 95% for any of the datasets.

In Arathi Krishna et Al’s “Phishing Detection using Machine Learning based URL Analysis: A Survey”, a thorough literature survey has been conducted on all the top publications, along with a summary of the different features that are extracted. The top features used are address bar features, abnormal based features, HTML based features and domain based features. 97.36% is the highest reported accuracy on the UCI dataset, where Random Forest is the most robust. However, they don’t go into detail with MLP-based approaches. In MN Alam et. Al’s “Phishing attacks detection using machine learning approach, there has been usage of a PCA-style feature selection technique before the application of random forest and decision tree classifiers. Since random forest classifier has less variance, they have observed that it might manage the over-fitting issue. They also observed that the random forest classifier delivers an accuracy of 97%. However, their limitation lies with the fact that exploration of a wide variety of features that aren’t restricted to domain-features is lacking.

2.3 Problem Statement Definition

Common users who look for information on the web require a method to ensure clicked links are secure since scams are prevalent and our tool aims to give users a prediction on whether the site is safe to visit.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

This visual representation can help a viewer understand the target consumer's point of view in terms of emotions, hearing amongst other senses. For this project, critical feelings like the feeling of being scammed have been listed along with the major user gains and pains/.

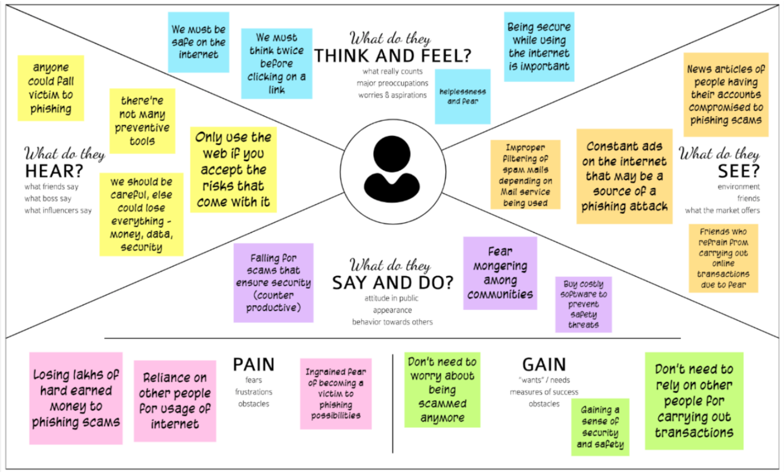
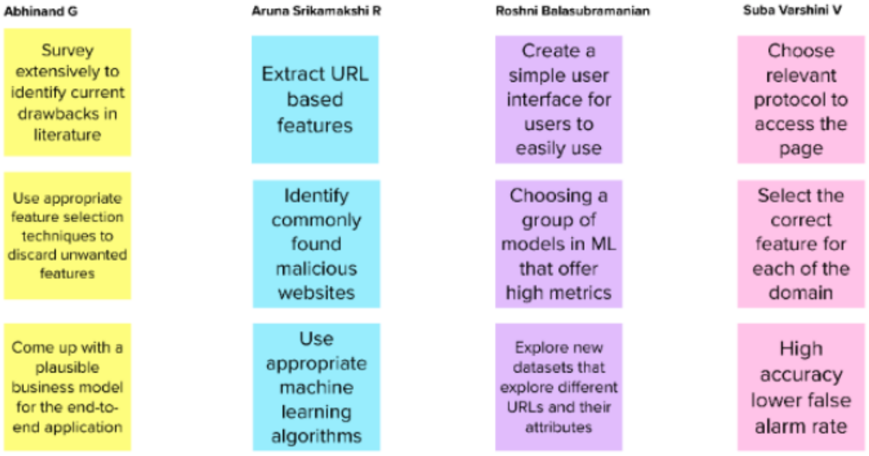

What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
We must be safe on the internet
We must think twice before clicking on a link
Being secure while using the internet is important
helplessness and fear
What do they
HEAR?
what friends say
what boss say
what influencers say
anyone could fall victim to phishing
there're not many preventive tools
Only use the web if you accept the risks that come with it
we should be careful, else could lose everything - money, data, security
What do they
SEE?
environment
friends
what the market offers
News articles of people having their accounts compromised to phishing scams
Improper filtering of spam mails depending on Mail service being used
Constant ads on the internet that may be a source of a phishing attack
Friends who refrain from carrying out online transactions due to fear
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
Falling for scams that ensure security (counter productive)
Fear mongering among communities
Buy costly software to prevent safety threats
PAIN
fears
frustrations
obstacles
Losing lakhs of hard earned money to phishing scams
Reliance on other people for usage of internet
Ingrained fear of becoming a victim to phishing possibilities
GAIN
"wants" / needs
measures of success
obstacles
Don't need to worry about being scammed anymore
Gaining a sense of security and safety
Don't need to rely on other people for carrying out transactions

Figure 3.1.1: Empathy Map Canvas

3.2 Ideation & Brainstorming

This step was very important for the team to formulate the different methods in which the pains of the customer can be eradicated and different ways in which a multitude of gains can be accomplished. Our team had a more technical and data science approach to solving the given problem. At first we decided to build a model but then realized that accuracy must be made a priority in order to minimize false negatives and false positives in the results. It was also made clear that the users must have an interface using which they can enter the URL and receive a prediction.



Abhinand G	Aruna Srikamakshi R	Roshni Balasubramanian	Suba Varshini V
Survey extensively to identify current drawbacks in literature	Extract URL based features	Create a simple user interface for users to easily use	Choose relevant protocol to access the page
Use appropriate feature selection techniques to discard unwanted features	Identify commonly found malicious websites	Choosing a group of models in ML that offer high metrics	Select the correct feature for each of the domain
Come up with a plausible business model for the end-to-end application	Use appropriate machine learning algorithms	Explore new datasets that explore different URLs and their attributes	High accuracy lower false alarm rate

Figure 3.2.1: Brainstorming

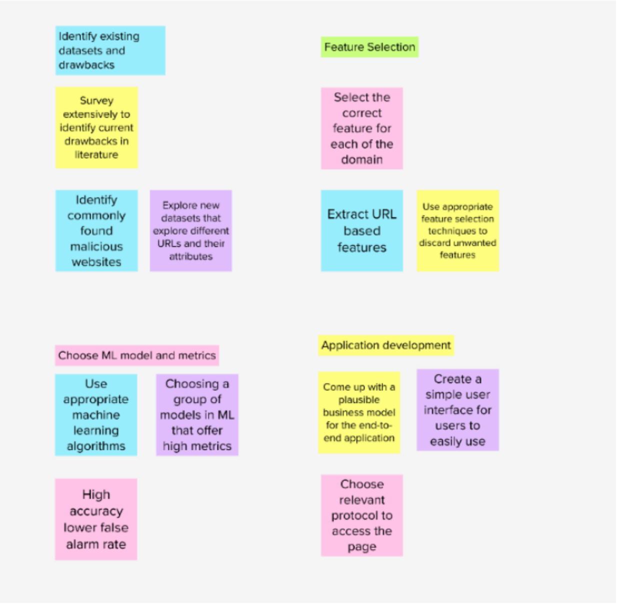
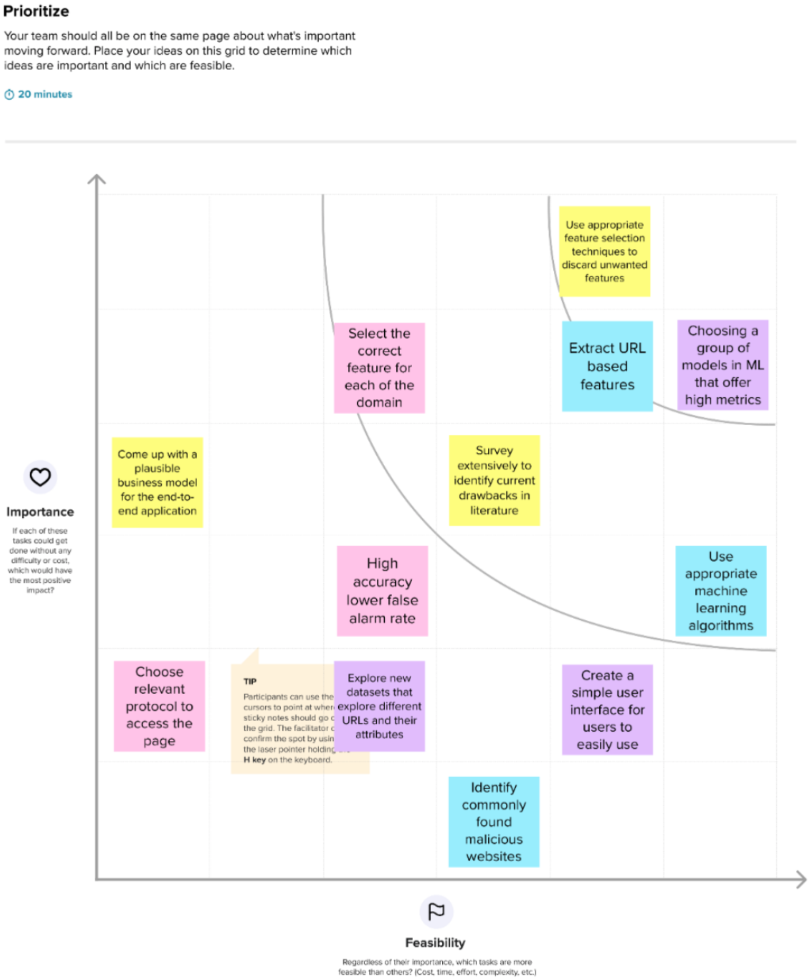

Identify existing datasets and drawbacks
Survey extensively to identify current drawbacks in literature
Identify commonly found malicious websites
Explore new datasets that explore different URLs and their attributes
Feature Selection
Select the correct feature for each of the domain
Extract URL based features
Use appropriate feature selection techniques to discard unwanted features
Choose ML model and metrics
Use appropriate machine learning algorithms
Choosing a group of models in ML that offer high metrics
High accuracy lower false alarm rate
Application development
Come up with a plausible business model for the end-to-end application
Create a simple user interface for users to easily use
Choose relevant protocol to access the page

Figure 3.2.2: Group Ideas



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

Use appropriate feature selection techniques to discard unwanted features
Select the correct feature for each of the domain
Extract URL based features
Choosing a group of models in ML that offer high metrics
Come up with a plausible business model for the end-to-end application
Survey extensively to identify current drawbacks in literature
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
High accuracy lower false alarm rate
Use appropriate machine learning algorithms
Choose relevant protocol to access the page
TIP
Participants can use the cursors to point at wher sticky notes should go c the grid. The facilitator c confirm the spot by usin the laser pointer holding H key on the keyboard.
Explore new datasets that explore different URLs and their attributes
Create a simple user interface for users to easily use
Identify commonly found malicious websites
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

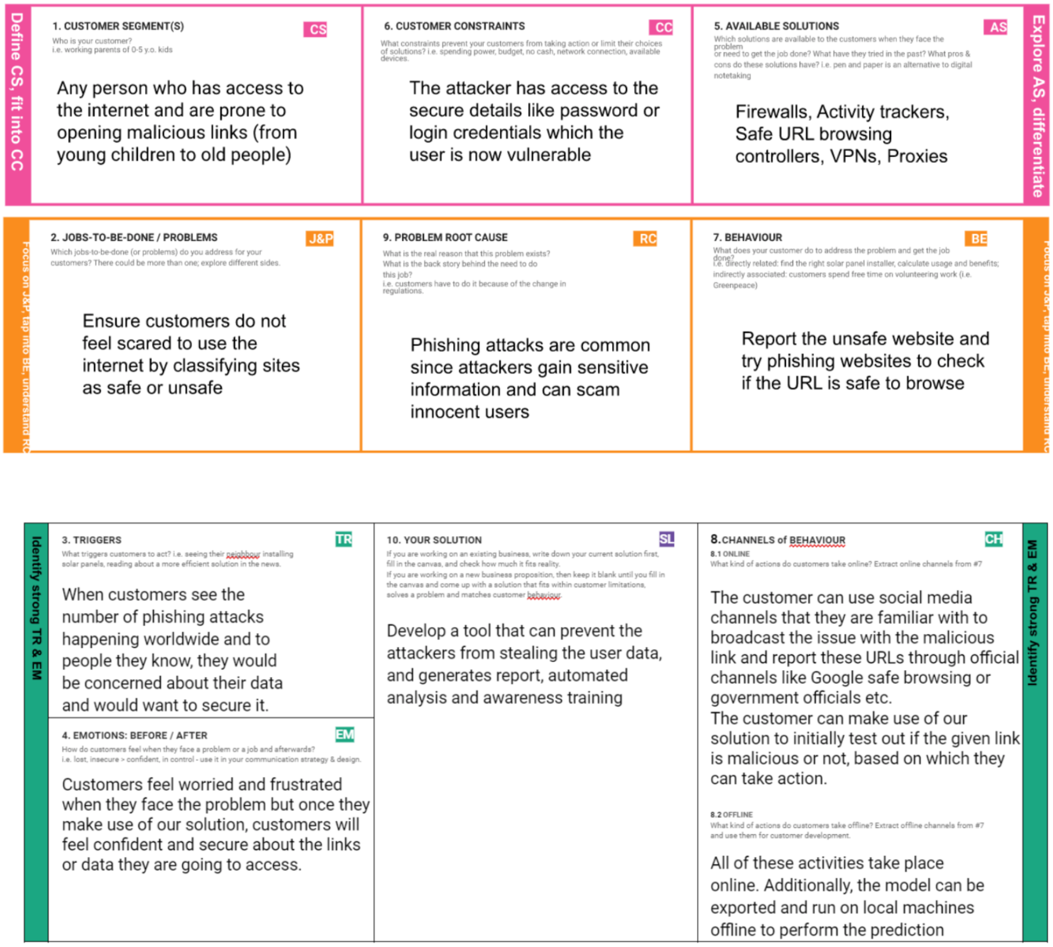
Figure 3.2.3: Prioritization

3.3 Proposed Solution

Table 3.3.1: Proposed Solution along with Description

S.No.	Parameter	Description
1.	**Problem Statement (Problem to be solved)**	To reduce the people falling for web phishing scams by creating a sophisticated tool that classifies a website as malicious or safe to use
2.	**Idea / Solution description**	Identify web phishing, classify whether it is an attack and prevent malicious intrusive websites
3.	**Novelty / Uniqueness**	● Uses an Ensemble model ● Extensive feature extraction strategy from the URL ● Simple, Easy-to-Understand UI
4.	**Social Impact / Customer Satisfaction**	● Users need not fear of losing lakhs of hard earned money to phishing scams & Users need not feel scared to use the internet ● Primarily targets the benefit of senior citizens and technologically challenged sections of the society ● Customers don't need to rely on offline transactions because of the fear of initiating transactions online
5.	**Business Model (Revenue Model)**	● B2B (Machine Learning model/API can be sold to various companies for their employees) and B2C Model (End product sold to individuals such as children's devices and senior citizens prone to attacks) ● Site can charge a one time fee for a device/user based on demographic surveys (Rs. 50 per year) ● Companies can be charged a discounted fee due to bulk purchase of the Application Programming Interface (API) ● Premium users will have access to details of the URL and reasonings for why a site has been classified 'unsafe'
6.	**Scalability of the Solution**	● Solution can use additional hardware resources when the amount of users and activity is increased ● The API can ensure that multiple requests at the same time are handled in a parallel fashion

3.4 Problem Solution fit


Define CS, fit into CC
Explore AS, differentiate
Focus on J&P, tap into BE, understand RC
Identify strong TR & EM
CS
CC
AS
J&P
RC
BE
TR
SL
CH
EM

1. CUSTOMER SEGMENT(S)
Who is your customer?
i.e. working parents of 0-5 y.o. kids

Any person who has access to the internet and are prone to opening malicious links (from young children to old people)

6. CUSTOMER CONSTRAINTS
What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

The attacker has access to the secure details like password or login credentials which the user is now vulnerable

5. AVAILABLE SOLUTIONS
Which solutions are available to the customers when they face the problem
or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Firewalls, Activity trackers, Safe URL browsing controllers, VPNs, Proxies

2. JOBS-TO-BE-DONE / PROBLEMS
Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

Ensure customers do not feel scared to use the internet by classifying sites as safe or unsafe

9. PROBLEM ROOT CAUSE
What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

Phishing attacks are common since attackers gain sensitive information and can scam innocent users

7. BEHAVIOUR
What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Report the unsafe website and try phishing websites to check if the URL is safe to browse

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

When customers see the number of phishing attacks happening worldwide and to people they know, they would be concerned about their data and would want to secure it.

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

Customers feel worried and frustrated when they face the problem but once they make use of our solution, customers will feel confident and secure about the links or data they are going to access.

10. YOUR SOLUTION
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.

Develop a tool that can prevent the attackers from stealing the user data, and generates report, automated analysis and awareness training

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

The customer can use social media channels that they are familiar with to broadcast the issue with the malicious link and report these URLs through official channels like Google safe browsing or government officials etc.
The customer can make use of our solution to initially test out if the given link is malicious or not, based on which they can take action.

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

All of these activities take place online. Additionally, the model can be exported and run on local machines offline to perform the prediction

Figure 3.4.1: Problem Solution Fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution.

Table 4.1.1: Functional Requirements

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Google and Email
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Input	User enters the suspicious URL in required field for validation
FR-4	URL Processing	By using appropriate machine learning algorithms and the the dataset, the model will process the new input
FR-5	Classification	The URL will be identified as Malicious or not

FR-6	Result	Result predicted by the model is displayed to the user will be cautioned about the website and the next steps to take if the URL turns out to be malicious

4.2 Non-Functional requirements

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

Table 4.2.1: Non-Functional Requirements

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	With an efficient, hassle-free, user friendly UI, users will not have difficulty in using the solution and navigating through the system
NFR-2	**Security**	Using Google authentication which automatically provides multi factor authentication and also authentication through email
NFR-3	**Reliability**	Probability of failure free operations in the specified environment of usage
NFR-4	**Performance**	The performance should be faster and user friendly for efficiency and effectivity
NFR-5	**Availability**	The model should be available for use always, it can be exported to users and can be run in the local machine
NFR-6	**Scalability**	This can be developed into an API, which can be incorporated by others who can make use of it

5. PROJECT DESIGN

5.1 Data Flow Diagrams

Information flow through a process or system is depicted in a data flow diagram. Data inputs, data outputs, data repositories, and the numerous subprocesses that the data goes through are all included. To represent multiple entities and their relationships, DFDs are constructed using standardised symbols and terminology.

Systems and processes that are challenging to convey in words are represented visually in data flow diagrams. These diagrams may be used to map out an existing system and improve it or to create and plan out a new system. It is simple to spot inefficiencies and create the optimum system by visualising each component.

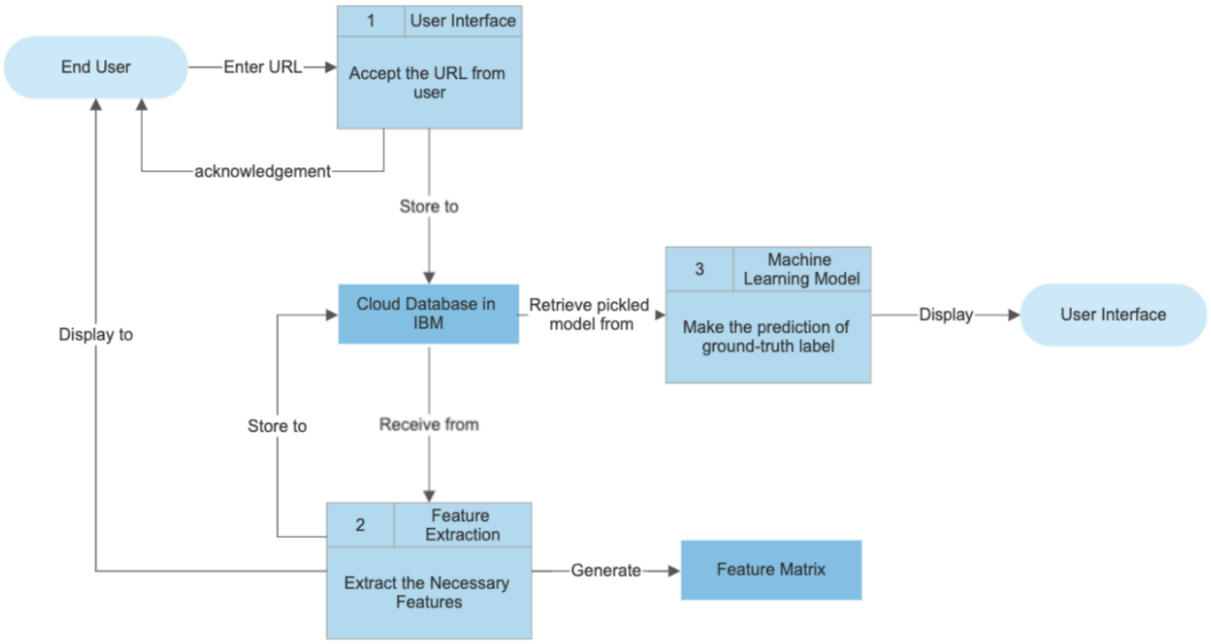


Figure 5.1.1: Data Flow Diagram

5.2 Solution & Technical Architecture

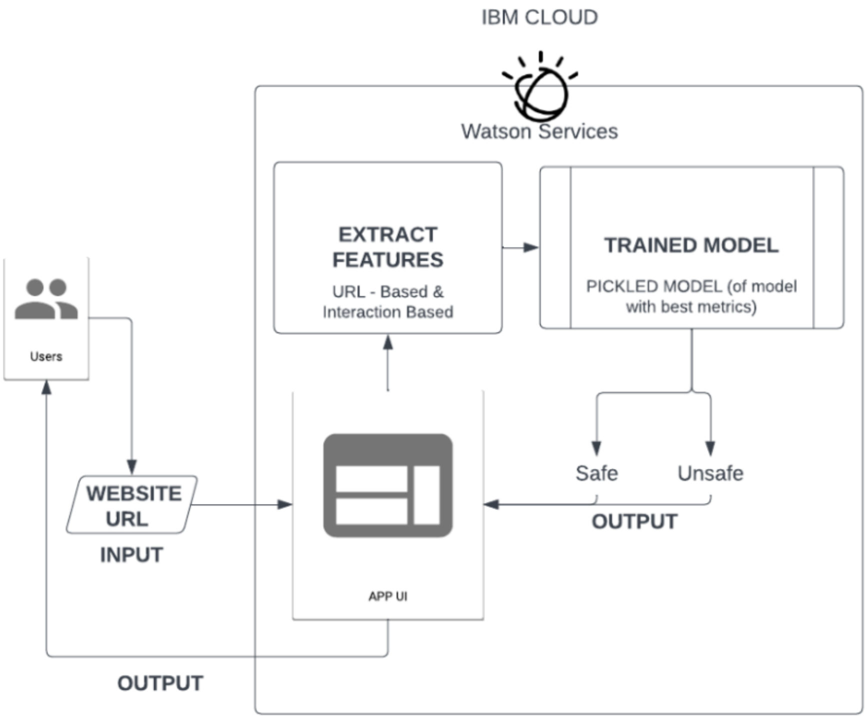


Figure 5.2.1: Technical Architecture

The technical architecture is composed of multiple modules. The User Interface component describes how the user interacts with the web UI. It contains both Input and Output components to it, where the user can input a URL and the output is displayed via the user interface. Additionally, it can also be emailed to the user if they have registered via email. The machine learning pipeline involves steps such as feature extraction, training of the model and selecting the best model. For the cloud infrastructure, the IBM cloud is made use of.

5.3 User Stories

The smallest piece of work in an agile system is a user story. It is a final objective, not a feature, as seen through the eyes of a software user. A user narrative is a casual, all-inclusive description of a software feature written from the viewpoint of the client or end user. A user story's objective is to describe how a piece of work will provide the client with a certain value. Keep in mind that "customers" don't always have to be end users on the outside in the conventional sense; they might be colleagues or internal customers within your company who depend on your team. User stories are short, straightforward statements that describe the desired result. They don't get specific. Later requirements are added. These have been explicated in Table 5.3.1.

Table 5.3.1: User Stories

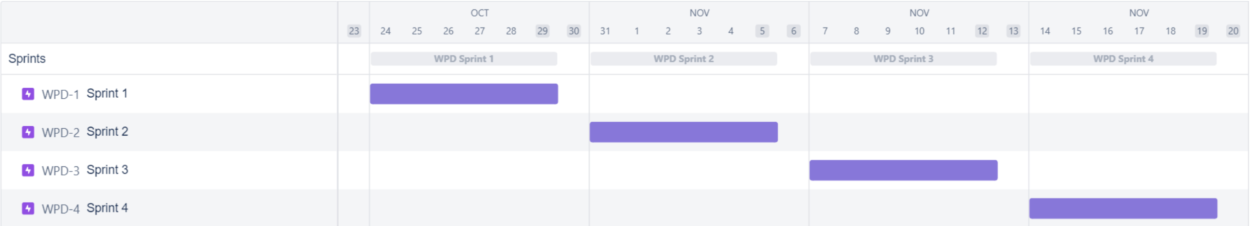
Sprint	**User Story Number**	**User Story / Task**	**Story Points**
Sprint-1	USN-5	As a user, I can view the training dataset and all the visualisation techniques on it	20
Sprint-2	USN-6	As a user, I can extract features from the suspicious URL after Pre-Processing	10
Sprint-2	USN-7	As a user, I can make use of machine learning models and receive a ground truth from the selected feature matrix after selecting the required features	10
Sprint-3	USN-8	As a user, I can check if my URL is malicious or not	20
Sprint-4	USN-2	As a user, I can log into the application by entering your email ID	10
Sprint-4	USN-4	As a user, I will receive a confirmation email to my email ID if the website I'm trying to check is malicious	10

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

As explained in Table 5.3.1, the sprints 1-4 have been planned with their corresponding user story numbers. Story points have been allocated to each user story, along with a description of the story and priority. A total of 80 points have been allocated, with 20 points belonging to each particular sprint. Each sprint focuses on a specific goal. Sprint 1 primarily focuses on the training of the dataset and performing exploratory data analysis techniques in order to gain a good understanding of the dataset. Sprint 2 focuses on pre-processing of the URL in order to extract features from it, along with making use of machine learning models. Sprint 3 focuses on selecting the best machine learning model and predicting whether any given test URL is malicious. Sprint 4 finally focuses on the end-to-end application with a friendly UI, that also has capabilities to send an email to the user after a prediction of the result.

6.2 Reports from JIRA



7. CODING & SOLUTIONING

7.1 Machine Learning

The machine learning solution focuses on two primary steps - Feature Extraction and Model Selection. In order to extract features, every particular function focuses on a separate feature. To quote a few examples, the following code uses the google-search python module in order to find the index of the website -

```
def GoogleIndex(self):
        try:
            site = search(self.url, 5)
            if site:
                return 1
            else:
                return -1
        except:
            return 1
```

Accordingly, either “1” or “-1” is entered into this particular column.

```
def AgeofDomain(self):
    try:
        creation_date = self.whois_response.creation_date
        try:
            if(len(creation_date)):
                creation_date = creation_date[0]
        except:
            pass
        today  = date.today()
        if age >=6:
            return 1
        return -1
    except:
        return -1
```

The whois library is used to calculate the age of the domain that has been entered by the user. In order to calculate the age, the year of creation is multiplied with 12 and added to the month of creation. If the result is greater than or equal to 6, 1 is passed as the feature. For other cases, the feature is set to -1. Once the feature extraction module has been completed, the training and testing sets have been identified using `train_test_split.` The best model is with the "Random Forest Classifier", where the number of estimators have been set to 100. The snippet used to generating the training and testing sets, and to train the model is as follows -

```
#Splitting into Training and Testing Sets
X_train,                          X_test,                          y_train,
y_test=train_test_split(X,y,train_size=0.7, shuffle=True)

#Machine Learning Model
rfc=RandomForestClassifier(n_estimators=100)
rfc.fit(X_train,y_train)
y_pred_rfc=rfc.predict(X_test)
```

Finally, to test the machine learning models on random URLs, the following code has been used -

```
rfc.predict(np.array(FeatureExtraction('https://www.testURL.com
').getFeaturesList()).reshape(1, -1))
```

7.2 Mail Server

The web application was developed using the following tools: JavaScript, HTML, IBM Cloud, and Flask. The programme accepts user URL inputs using HTTP POST requests. In order to deliver the prediction results when a user visits a malicious website, the SMTP server for Gmail communicates with the user's email address. The port that has been used is `465`, and the email is sent under the username `webphishingdetection@gmail.com.`

```
mail = Mail(app)
app.config['MAIL_SERVER'] = 'smtp.gmail.com'
app.config['MAIL_PORT'] = 465
app.config['MAIL_USERNAME'] = 'webphishingdetection@gmail.com'
app.config['MAIL_PASSWORD'] = ''
app.config['MAIL_USE_TLS'] = False
app.config['MAIL_USE_SSL'] = True
mail = Mail(app)
```

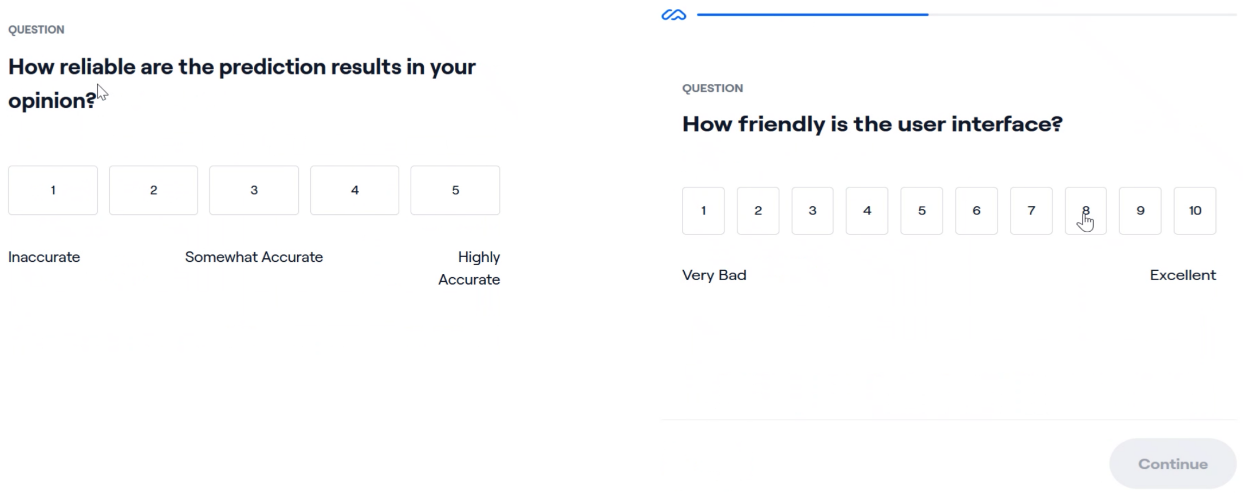
```
msg = Message(
                'Web Phishing Detection - Identified Maliciour URL',
                 sender='webphishingdetection@gmail.com',
                 recipients=[email] )
                 msg.body = 'This url: '+url + \
                  ' is malicious as per our predictions, please be careful
and not open the website'
            mail.send(msg)
```

As observed in the above code snippets, the mail server has been configured using parameters like `MAIL_PASSWORD, MAIL_USE_TLS, MAIL_USE_SSL, MAIL_PORT`, and so on. The message body of the email has also been configured and customized regarding the application's requirements.

8. TESTING

8.1 User Acceptance Testing

Before deploying the software application to a production environment, the end user or client performs a type of testing known as user acceptance testing, or UAT. After functional, integration, and system testing are complete, UAT is carried out as the last stage of testing. To perform UAT in our case, the tool "Maze" has been used, where multiple questions are asked in the form of a survey to understand the usability from a client's perspective. An example of the questions posed to the client with respect to the end application's usability has been depicted in Figure 8.2.1. The dynamic visualisation of the clients' report has been given in Figure 8.2.2, where it is easy for the developers' to modify modules accordingly to satisfy user requirements.



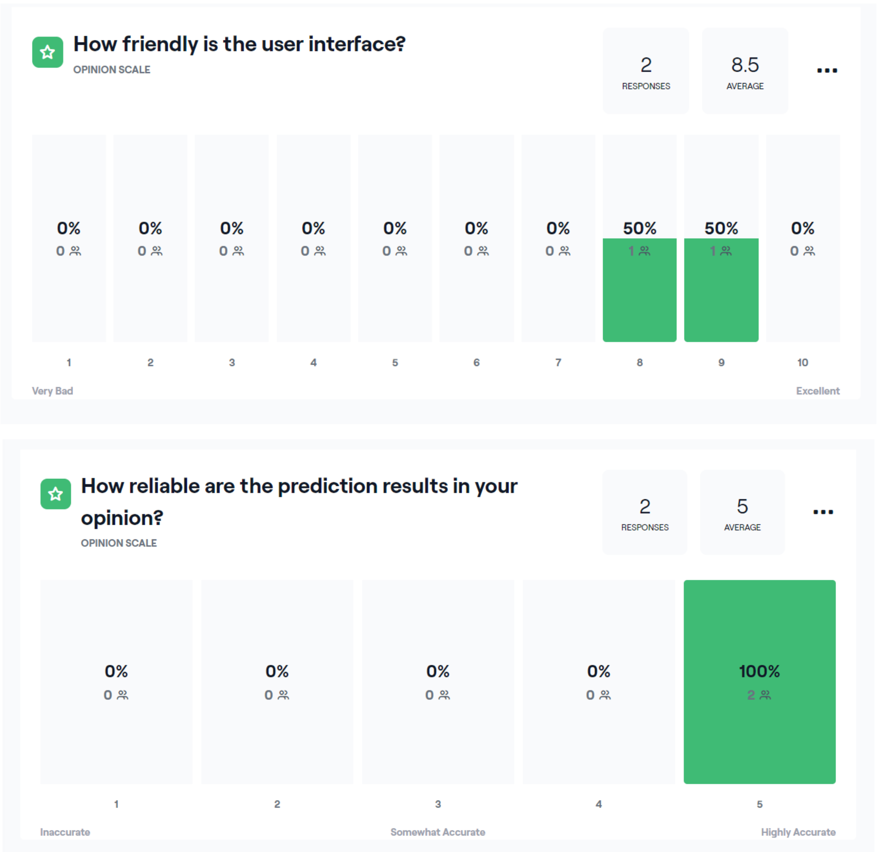
QUESTION

How reliable are the prediction results in your opinion?

QUESTION

How friendly is the user interface?

Figure 8.2.1: Survey for User Acceptance Testing in Maze



How friendly is the user interface?

OPINION SCALE

2 RESPONSES

8.5 AVERAGE

How reliable are the prediction results in your opinion?

OPINION SCALE

2 RESPONSES

5 AVERAGE

Figure 8.2.2: Visualization of Results from UAT

8.2 Defect Analysis

Table 8.2.1: Defect Analysis Table

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	1	17
Duplicate	1	0	0	0	1
External	2	3	0	1	6
Fixed	11	2	4	14	31
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	10	18	66

8.3 Test Case Analysis

Table 8.3.1: Test Case Analysis Table

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

9. RESULTS

9.1 Performance Metrics

S.No	Classifier	Number of Estimators	Accuracy	F1-Score	Recall
1	Random Forest	10000	97%	97% for -1 98% for 1	96% for -1 98% for 1
2	Random Forest	500	97%	97% for -1 98% for 1	96% for -1 98% for 1
3	Random Forest	100	98%	97% for -1 98% for 1	97% for -1 98% for 1

10. ADVANTAGES & DISADVANTAGES

The major advantages that the project brings is the reduction of fear and enhanced security while using the web. In today's web surfing and the amount of hazards that can occur, it is essential to introduce tools with high accuracy to overcome the potential of getting scammed or phished. Hence, our advantage is producing a tool that can predict whether the site is safe. The disadvantage to the project is that it is possible for false positives and false negatives are not impossible. The accuracy metric is 98%. Hence, there is a sliver of a chance to get false results.

11. CONCLUSION

Taking into account the fear that exists within the users of the web, the team has created a tool to enable safe surfing by entering the website URL in order to receive a prediction. The team has undergone brainstorming and ideation as well as stepping into the shoes of the ideal user. After brainstorming and prioritization, a final sprint plan was developed, after which the team worked on developing and pushing the outputs at each stage. After data analysis, model building, interface developing, cloud deployment and testing, the final tool has been created that the public can use.

12. FUTURE SCOPE

The project can be further extended by implementing deep learning and comparing the effectiveness with the current result. Additionally, the scalability of the project can be improvised by accommodating multiple users at the same time. End to end application deployment and commercial rendering of the website as a whole can be furthermore researched upon.

13. APPENDIX

Source Code

GitHub - https://github.com/IBM-EPBL/IBM-Project-1205-1658378450

Project Demo Link: Click Here to View Demo