Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	03 October 2022
Team ID	PNT2022TMID46888
Project Name	Project –Web Phishing Detection
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

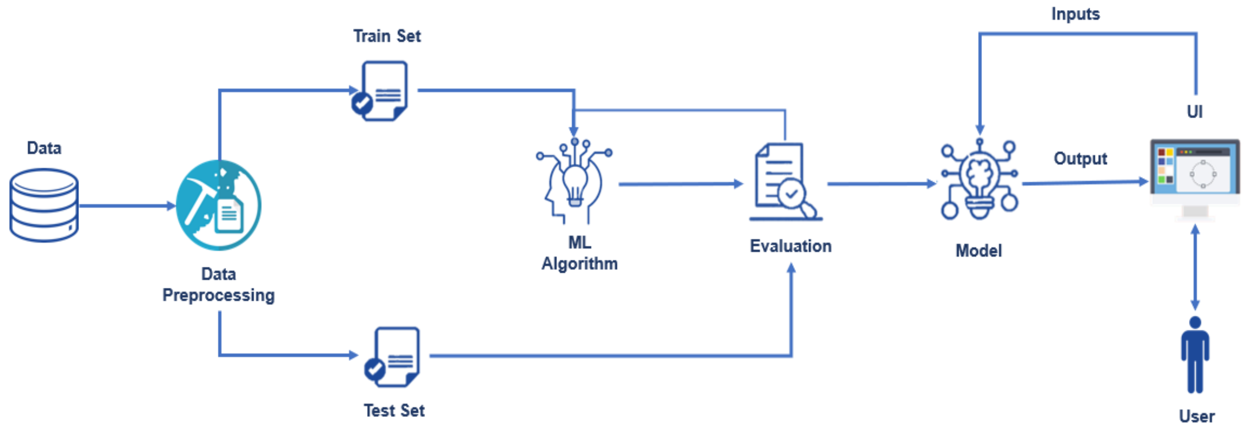

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	UI incure extra cost for the computation of each authentication and may also require users to keep extra authentication devices making it cumbersome to implement and use	Machine learning, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	URL, HTTP header, host and web content, and the network layer features such as remote server attributes, crawler-server communication attributes, and DNS information, and aggregated these features to train a variety of	Java / Python

		classification models to detect phishing websites	
3.	Application Logic-2	applications grow over the globe and it becomes an inseparable part of modern life; scams and fraudulent also join to the users of it.	IBM Watson STT service
4.	Application Logic-3	Although, in this experiment it was presumed that the false positive and false negative is equally important. This can be correct due to the application.	IBM Watson Assistant
5.	Database	The database receives thousands of potential phishing URLs every day from various sources. The server in turn makes the request to UAB's database to send new a list of fresh data. If any new URLs are available, the database will send them to the server, which will in turn forward them to the appropriate client	MySQL, NoSQL, etc.
6.	Cloud Database	Amazon the micro instance, has sufficient computing resources for this type of fetching client. The I/O used for this experiment was averaged for a per day usage and projected over a 30 day period. This price could be lowered by as much as 40% if the parsing was done as the data was retrieved and only the parsed results sent to the control server.	IBM DB2, IBM Cloud and etc.
7.	File Storage	The system will record each run's parameters and data collection types and dump the model to the file storage system.	IBM Block Storage or Other Storage Service or Local File system

8.	External API-1	For each such URL, an HTTP API request is sent, and the response to this request is used to detect phishing. The Firefox web browser also uses the Google Safe Browsing API for phishing detection. It alerts the user if	IBM Weather API, etc.
9.	External API-2	API results indicate phishing. Similar to it also uses google safe browsing API but the results demonstrate that it gives faster response time.	Aadhar API, etc.
10.	Machine Learning Model	The machine learning module is mainly responsible for model training and model testing. In this framework, the data to extract features. We imported a scikit-learn library	Logistic Regression, Random Forest, and support vector machine.
11.	Infrastructure (Server / Cloud)	The costs of maintaining a full time cloud based fetching instance was calculated to determine the feasibility of its use in a real world application.	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Open source deep learning framework and development platform we used to the dataset module to build a custom datasets input for the training model	PYTORCH
2.	Security Implementations	List all the security / access controls implemented, User opens the web browser and opens the email on the web browser. The email before opening will be scanned by the backend	Spoofing detection, fraud detection, filtering/blocking technology

S.No	Characteristics	Description	Technology
		phishing detection engine. The ‘visible _links’ will be extracted from the email body. The “invisible _links” will be extracted from the email body. The “Unmatching_urls” will be extracted from the email body	
3.	Scalable Architecture	detect and isolate both phishing e-mail senders and phishing web servers. In addition, we propose to develop a self-management architecture that enables ISPs to protect their users against phishing attacks,	Machine learning algorithm
4.	Availability	This service will be available on laptops, tablets and mobile devices	Evaluation training dataset, Data pre processing.
5.	Performance	The system should be fast and accurate System will handle expected and non-expected errors in a manner that will prevent information loss and long downtime period. System should be able to handle large amounts of data, System should accommodate high number of photos and users without any fault	Securing the user data