**AN ENGINEERING PROJECT REPORT**

**ON**

**“****Phishing URL detection using ML”**

**Submitted By**

**Sushil Paudel – 200348**

**Bibek Pandeya – 200311**

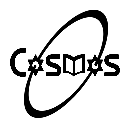
**Bhushan Bartaula – 200310**

**Prakash Lamichhane – 200323**

**Submitted To**

**Department of IT and Computer Engineering**

**In partial fulfillment of requirement for the degree of Bachelor of engineering in Computer Engineering.**



**Cosmos College of Management & Technology**

**(Affiliated with Pokhara University)**

**Tutepani, Lalitpur, Nepal**

**Date of Submission: - 2082/04/05**

# **DECLARATION**

We hereby declare that the report of the project entitled “**Phishing URL Detection using ML**” which is being submitted to the Department of ICT, Cosmos College of Management and Technology, Tutepani-14, Lalitpur in the partial fulfillment of the requirements for the award of the Degree of Bachelor of Engineering in Computer Engineering, is a Bonafide report of the work carried out by us. The materials contained in this report have not been submitted to any University or Institution for the award of any degree and we are the only author of this complete work and no sources other than the listed here have been used in this word.

**Sushil Paudel - 200348**

**Bibek Pandeya - 200311**

**Bhushan Bartaula - 200310**

**Prakash Lamichhane - 200323**

# **CERTIFICATE OF APPROVAL**

The project report entitled **“Phishing URL Detection using ML”**, submitted by Sushil Paudel, Bibek Pandeya, Bhushan Bartaula and Prakash Lamichhane in partial fulfillment of the requirement for the Bachelor's degree in Computer Engineering has been accepted as a bona fide record of work independently carried out by the group in the department.

……………………………..

**External Examiner**

Name:…………………. Designation:

……………

Affiliation: ………………….

………………………………

**Project Supervisor**

Er. Ajaya Adhikari

HOD, Department of ICT

Cosmos College of Management and Technology

**COPYRIGHT**

The author has agreed that the library, **Cosmos College of Management and Technology**, Tutepani-14, Lalitpur, may make this report freely available for inspection. Moreover, the author has agreed that permission for extensive copying of this project report for scholarly purpose may be granted by the lecturers, who supervised the project works recorded herein or, in their absence, by the Head of Department wherein the project report was done. It is understood that the recognition will be given to the author of the report and to the Department of ICT, CCMT, in any use of the material of this project report. Copying or publication or other use of this report for financial gain without the approval of the Department and author’s written permission is prohibited. Request for permission to copy or to make any other use of the material in this report as a whole or in part should be addressed to the Head of Department, Department of Information and Communication Technology.

# **ACKNOWLEDGEMENT**

We would like to express our heartfelt gratitude to all those who supported and guided us throughout the journey of this project. First and foremost, we extend our sincere thanks to our **supervisor Er.Ajaya Adhikari**, whose invaluable support, timely feedback, and continuous encouragement played a crucial role during the implementation and successful completion of this project.

We are also deeply thankful to the **Department of Information and Communications Technology, Cosmos College of Management and Technology, Tutepani, Lalitpur** for providing us with the necessary consultation, knowledge, and academic environment that made this project possible.

Our special thanks go to all the faculty members for their motivation, guidance, and encouragement. Their insightful suggestions and constructive feedback were instrumental in helping us refine our work.

Lastly, we warmly welcome any constructive criticism and suggestions for further improvement of this project.

**Sushil Paudel**

**Bibek Pandeya**

**Bhushan Bartaula**

**Prakash Lamichhane**

# **ABSTRACT**

Phishing attacks continue to evolve as a major cybersecurity threat, resulting in significant financial and data losses for individuals and organizations worldwide. These attacks typically rely on deceptive tactics to trick users into revealing sensitive information such as usernames, passwords, and financial credentials. Common phishing vectors include emails, instant messages, pop-up alerts, and fraudulent websites.

This project presents a machine learning-based approach to detect phishing websites by classifying URLs as either legitimate or malicious. The system is trained on a comprehensive dataset containing both phishing and benign URLs, collected from publicly available threat intelligence sources and academic datasets.

To improve prediction accuracy, multiple machine learning models, including deep neural networks, are employed and evaluated. The dataset, consisting of over 5,000 raw URLs, is expanded into a total of 10,000 samples, split into 80% training and 20% testing data. The features used for classification are grouped into three categories: **address bar-based features**, **domain-based features**, and **HTML & JavaScript-based features**.

As a practical implementation, a web application is developed that allows users to input any URL and receive real-time feedback on whether the link is safe or potentially harmful. This system aims to contribute to safer internet browsing by providing an intelligent and accessible phishing detection tool.

Table of Contents

[**DECLARATION** ii](#_Toc204003013)

[**CERTIFICATE OF APPROVAL** iii](#_Toc204003014)

[**ACKNOWLEDGEMENT** v](#_Toc204003015)

[**ABSTRACT** vi](#_Toc204003016)

[**LISTS** viii](#_Toc204003017)

[List of Figures: viii](#_Toc204003018)

[List of Tables: - viii](#_Toc204003019)

[List of abbreviations: - viii](#_Toc204003020)

[**CHAPTER 1: INTRODUCTION** 1](#_Toc204003021)

[1.1 Background Information: 1](#_Toc204003022)

[1.2 Statements regarding the problems: 2](#_Toc204003023)

[1.3 Scope of the study: 2](#_Toc204003024)

[1.4 Objectives: 2](#_Toc204003025)

[1.5 Methodology: 2](#_Toc204003026)

[1.5.1 Data Collection: 3](#_Toc204003027)

[1.5.2 Data Preprocessing: 3](#_Toc204003028)

[1.5.3 Feature Extraction: 3](#_Toc204003029)

[1.5.4 Model Implementation: 3](#_Toc204003030)

[1.5.5 Model Evaluation: 4](#_Toc204003031)

[1.5.6 Web Application Development: 4](#_Toc204003032)

[**1.6 Significance of the Project:** 4](#_Toc204003033)

[**CHAPTER 2: LITERATURE REVIEW** 5](#_Toc204003034)

[**2.1 Overview of the Study** 5](#_Toc204003035)

[**2.2 Theoretical Review:** 5](#_Toc204003036)

[**CHAPTER 3: SYSTEM ANALYSIS AND DESIGN** 8](#_Toc204003037)

[**3.2 Overview of System Analysis** 8](#_Toc204003038)

[**3.2 Analysis of Existing System** 8](#_Toc204003039)

[**3.3 Proposed System** 9](#_Toc204003040)

[**3.3.1 Benefits of the new system** 9](#_Toc204003041)

[**3.4 Model Development Model** 9](#_Toc204003042)

[i. Data Processing 9](#_Toc204003043)

[ii. Preprocessing 10](#_Toc204003044)

[Exploratory Data Analysis 10](#_Toc204003045)

[Feature Extraction 10](#_Toc204003046)

[Model Training 11](#_Toc204003047)

[Model Testing 11](#_Toc204003048)

[Model Evaluation 12](#_Toc204003049)

[**3.5 System Modelling** 12](#_Toc204003050)

[3.5.1 System Architecture 12](#_Toc204003051)

[3.5.2 Use a case diagram of the system 13](#_Toc204003052)

[3.5.3 Flowchart of the system 14](#_Toc204003053)

[**4.1 Software Process Model** 17](#_Toc204003054)

[8. OUTPUT: - 19](#_Toc204003055)

[**9. CONCLUSION: -** 22](#_Toc204003056)

[**10. BIBLIOGRAPHY: -** 23](#_Toc204003057)

# **LISTS**

# List of Figures:

[Figure 1 :Phases of Iterative Increment Model 8](#_Toc198884436)

[Figure 2: Architectural Design of the system 9](#_Toc198884437)

[Figure 3: Flowchart of the system 12](#_Toc198884438)

[Figure 4: Flowchart of the web interface 13](#_Toc198884439)

[Figure 5: Use case diagram of system 14](#_Toc198884440)

## List of Tables: -

## List of abbreviations: -

**URL** Uniform Resource Locators

**HTML** Hyper Text Markup Language

**EDA** Exploratory Data Analysis

**SVM** Support Vector Machine

**CSV** Comma Separated Values

**JSON** JavaScript Object Notation

**API** Application Programming Interface

**IDE** Integrated Development Environment

# **CHAPTER 1: INTRODUCTION**

## 1.1 Background Information:

The rapid growth of the internet and digital communication has significantly transformed the way individuals and organizations interact, share information, and conduct business. While this digital transformation has enabled countless opportunities, it has also opened the door to various cybersecurity threats, one of the most persistent and damaging being **phishing attacks**.

**Phishing** is a form of cybercrime that involves tricking individuals into revealing confidential information such as usernames, passwords, credit card numbers, and other sensitive data by impersonating a trustworthy entity in digital communication. Cybercriminals exploit various communication channels including emails, websites, social media platforms, and instant messaging services to carry out phishing attempts.

As phishing techniques become increasingly sophisticated, traditional rule-based detection methods often fail to provide adequate protection. In recent years, **machine learning** has emerged as a promising solution in the field of cybersecurity, offering the ability to analyze patterns, detect anomalies, and classify malicious content more accurately.

This project is motivated by the increasing need for **automated, intelligent solutions** that can detect phishing URLs with high accuracy and minimal human intervention. The goal is to develop a machine learning-based system capable of identifying phishing websites based on a wide range of features extracted from URLs. By training and testing the model on a robust dataset consisting of both legitimate and phishing URLs and incorporating different types of feature sets (such as address bar-based, domain-based, and HTML/JavaScript-based), the system aims to provide reliable classification results.

Furthermore, a user-friendly **web application** is developed as part of the project, allowing end-users to input URLs and receive instant feedback on their legitimacy. This solution contributes to raising awareness and enhancing security among everyday internet users, ultimately reducing the risk posed by phishing threats.

## 1.2 Statements regarding the problems:

Phishing attacks have gotten increasingly complex; it is very difficult for an average person to determine if an email message link or website is legitimate. Cyber-attacks by criminals that employ phishing schemes are so prevalent and successful nowadays. Hence, this project seeks to address fake URLs and domain names by identifying phishing website links. Therefore, having a web application that provides the user with an interface to check if a URL is Phishing or legitimate will help decrease security risks to individuals and organizations.

## 1.3 Scope of the study:

This study explores data science and machine learning models that use datasets gotten from open-source platforms to analyze website links and distinguish between phishing and legitimate URL links.

The model is integrated into a web application, allowing a user to predict if a URL link is legitimate or phishing. This online application is compatible with a variety of browsers.

## 1.4 Objectives:

The main objective of our project is to develop a machine learning-based system capable of accurately detecting and classifying phishing URLs and deploying them as a web application for real-time usage. Besides the main objective, the specific objectives can be enlisted as:

1. Collect and preprocess a dataset of phishing and legitimate URLs.
2. Extract key features from URLs for classification.
3. Implement and evaluate multiple machine learning models.
4. Compare model performance based on accuracy and other metrics.
5. Develop a user-friendly web application for URL detection.
6. Promote phishing awareness through intelligent detection.

## 1.5 Methodology:

The development of the phishing URL detection system is carried out through a structured methodology involving data collection, preprocessing, feature extraction, model implementation, evaluation, and deployment. The entire process is designed to ensure that the model achieves high accuracy and usability.

### 1.5.1 Data Collection:

Datasets are gathered from publicly available sources containing both **phishing** and **legitimate** URLs. Sources include open phishing databases and academic repositories, ensuring a diverse and balanced dataset.

### 1.5.2 Data Preprocessing:

Duplicate and null entries are removed to maintain data quality. URLs are labeled as **"phishing"** or **"legitimate"** based on their source. The dataset is split into **training (80%)** and **testing (20%)** sets for model development and validation.

### 1.5.3 Feature Extraction:

Features are extracted from each URL based on three main categories:

1. **Address Bar-Based Features**: Length of the URL, presence of IP address, use of “" symbol, etc.
2. **Domain-Based Features**: Age of the domain, domain registration length, DNS record, etc.
3. **HTML & JavaScript-Based Features**: Use of suspicious scripts, iframe tags, and redirection behavior.

### 1.5.4 Model Implementation:

Multiple machine learning algorithms are implemented, including:

* Logistic Regression
* Decision Tree
* Random Forest
* Support Vector Machine (SVM)
* K-Nearest Neighbors (KNN)
* Naïve Bayes
* Deep Neural Networks (DNN)

Models are trained using the training dataset and validated using the testing dataset.

### 1.5.5 Model Evaluation:

Performance of each model is evaluated using the following metrics:

* **Accuracy**
* **False Positive Rate**
* **False Negative Rate**

Comparative analysis is performed to determine the most effective model for phishing URL detection.

### 1.5.6 Web Application Development:

A **web-based interface** is built using appropriate front-end and back-end technologies. Users can enter a URL to check its legitimacy in real time. The trained model is integrated into the backend to provide instant classification results.

## **1.6 Significance of the Project:**

This project plays an important role in addressing the growing threat of phishing attacks by using machine learning to detect malicious URLs. It promotes cybersecurity awareness and provides a practical tool for users to check the legitimacy of websites in real time. Academically, it demonstrates the application of machine learning in real-world scenarios and contributes to ongoing research in intelligent threat detection systems.

# **CHAPTER 2: LITERATURE REVIEW**

## **2.1 Overview of the Study**

This chapter offers an insight into various important studies conducted by excellent scholars from articles, books, and other sources relevant to the detection of phishing websites. It also provides the project with a theoretical review, conceptual review, and empirical review to demonstrate understanding of the project.

## **2.2 Theoretical Review:**

According to the [2024 Verizon DBIR](https://www.verizon.com/business/resources/reports/dbir/), the human element is contained in 68% of breaches. Of those, the [Comcast Business Cybersecurity Threat Report says 80-95%](https://corporate.comcast.com/press/releases/comcast-business-report-global-cybersecurity-technology-advancements-accelerate) are initiated by a phishing attack, and the total volume of phishing attacks has skyrocketed by 4,151% since the advent of ChatGPT in 2022, [according to SlashNext](https://slashnext.com/press-release/slashnext-mid-year-state-of-phishing-report-shows-341-increase-in-bec-and-advanced-phishing-attacks/).

At an estimated $4.88M per phishing breach ([IBM Cost of a Data Breach Report 2024](https://www.ibm.com/reports/data-breach)), social engineers are making billions by being better at making people click than we are at understanding what makes them tick.

A screenshot of a graph

AI-generated content may be incorrect.

The **RSA Anti-Fraud Command Center** reported a **160% increase in phishing attacks in 2012** compared to 2011, indicating a steep rise in cybercrime activities. In **2013**, the total number of phishing attacks rose to approximately **450,000**, with estimated **financial losses exceeding USD 5.9 billion**. These numbers underscore the seriousness of phishing threats as shown in Fig: 2.2.2.

A graph showing the number of losses

AI-generated content may be incorrect.

*Figure 2.1 Worldwide financial losses (in billion) due to phishing attacks*

According to the **APWG Phishing Activity Trends Report (2014)**, the **first quarter of 2014** recorded **125,215 phishing attacks**, marking a **10.7% increase** over the last quarter of 2013. Additionally, over **55% of phishing websites** included the name of the targeted organization to appear more trustworthy, and **99.4%** of phishing websites operated on **port 80**—the standard HTTP port—making them blend seamlessly with legitimate traffic as shown in Fig: 2.2.3.

A graph of the number of people in the same direction

AI-generated content may be incorrect.

*Figure 2.2 growth of phishing attacks from 2005 to 2015*

According to **Ankit and Gupta (2017)**, as cited in *Internet World Stats (2014)*, the global number of internet users reached **2.97 billion in 2014**, representing more than **38% of the world population**. The rapid digital expansion has provided attackers with a larger pool of potential victims, especially those unaware of phishing tactics.

# **CHAPTER 3: SYSTEM ANALYSIS AND DESIGN**

## **3.2 Overview of System Analysis**

This section presents a comprehensive overview of the analytical approach adopted throughout the project to meet its objectives. System analysis plays a critical role in understanding the problem domain, evaluating existing solutions, and designing a more efficient and robust system. It serves as the foundation upon which effective system development is built.

The methodology chosen for this research guided the process of identifying, analyzing, and solving the challenges associated with phishing URL detection. A clear understanding of the problem, combined with a structured analysis of existing systems and technologies, enabled the development of a more intelligent and user-friendly solution.

According to the **Merriam-Webster Dictionary**, system analysis is defined as the process of studying a procedure or operation to determine its goals and purposes, and to develop systems and procedures that achieve those goals efficiently. In the context of this project, system analysis involves examining how phishing detection systems operate, identifying their limitations, and exploring opportunities to enhance their accuracy and usability through machine learning.

By thoroughly analyzing both the problem and the existing methods, this project ensures that the proposed solution is not only technically sound but also practically applicable in real-world scenarios.

## **3.2 Analysis of Existing System**

The existing phishing detection systems often suffer from low accuracy and high false alarm rates, particularly when new and evolving phishing techniques are introduced. Most rely heavily on blacklist-based methods, which are ineffective against emerging threats. As registering new domains has become increasingly easy, no blacklist can remain fully up to date, making such approaches insufficient for reliable phishing detection.

## **3.3 Proposed System**

The proposed phishing detection system leverages machine learning models and deep neural networks, comprising two main components: the learning models and a web application. The models include Decision Tree, Support Vector Machine, XGBoost, Multilayer Perceptron, Autoencoder Neural Network, and Random Forest.

These models are chosen based on performance comparisons among various algorithms. Each model is trained and tested using content-based features extracted from both phishing and legitimate datasets.

Ultimately, the model with the highest accuracy is selected and integrated into a web application, allowing users to check whether a given URL is phishing or legitimate.

### **3.3.1 Benefits of the new system**

i. Capable of accurately distinguishing between phishing (0) and legitimate (1) URLs.  
ii. Helps minimize phishing-related data breaches within organizations.  
iii. Beneficial to both individuals and organizations in enhancing cybersecurity.  
iv. Designed with a user-friendly interface, making it simple and accessible for all users.

## **3.4 Model Development Model**

The model development method involves utilizing multiple machine learning models, evaluating their performance, and refining them through an iterative process until a model that meets the desired requirements is achieved. Figure 3.1 illustrates the systematic steps followed in developing these machine learning models, incorporating both supervised and unsupervised learning techniques.

The following are the major stages involved in developing the machine learning model for phishing detection systems:

### Data Processing

The datasets used for training the models were sourced from various open-source platforms. The dataset consists of both phishing and legitimate URLs. The phishing URLs were collected from an open-source service called *PhishTank*, which provides phishing URL lists in formats like CSV and JSON, updated hourly. From this dataset, over 5,000 phishing URLs were randomly collected.  
The legitimate URLs were obtained from the open datasets provided by the University of New Brunswick. This dataset includes benign, spam, phishing, malware, and defacement URLs. For this project, only benign URLs were considered, and over 5,000 of them were randomly selected to train the machine learning models.

### Preprocessing

Data preprocessing is a critical step following data collection. The raw dataset was cleaned by removing redundant and inconsistent data. The dataset was then encoded using the One-Hot Encoding technique to transform it into a suitable format for machine learning models.

### Exploratory Data Analysis

After preprocessing, exploratory data analysis techniques were applied to gain insights into the dataset. Visualization tools such as heatmaps, histograms, box plots, scatter plots, and pair plots were used to explore and summarize the data. These visualizations helped in identifying patterns, correlations, and outliers.

### Feature Extraction

The aim of feature extraction is to reduce dimensionality while preserving important information. Website content-based features were extracted from both phishing and legitimate URLs. These features include:

Address Bar-Based Features (9 features)

Domain-Based Features (4 features)

HTML & JavaScript-Based Features (4 features)  
In total, 17 features were extracted and used for phishing detection.

A diagram of a machine learning

AI-generated content may be incorrect.

*Fig 3.1: Machine Learning development Process*

### Model Training

Model training involves feeding machine learning algorithms with data to help identify and learn good attributes of the dataset.This research problem is a product of supervised learning, which falls under the classification problem. The algorithms used for phishing detection consist of supervised machine learning models (4) and deep neural networks (2), which were used to train the dataset. These algorithms include Decision Tree, Random Forest, Support Vector Machines, XGBooster, Multilayer Perceptron, and Auto-encoder Neural Network.

All these models were trained on the dataset. Thus, the dataset is split into a training and testing set. The training model consists of 80% of the dataset to enable the machine learning models to learn more about the data and be able to distinguish between phishing and legitimate URLs.

### Model Testing

Model Testing involves the process where the performance of a fully trained

model is evaluated on a testing set.

Thus, after 80% of data has been trained, 20% of the dataset is used to

evaluate the trained dataset to see the performance of the models.

### Model Evaluation

Model Evaluation involves estimating the generalization accuracy of models and deciding whether the model perform better or not. Thus, Scikit-learn (sklearn matrics) module was used to implement several score and utility functions to measure the classification performance to properly evaluate the models deployed for phishing detection.

## **3.5 System Modelling**

System modeling involves the process of developing an abstract model of a system, with each model presenting a different view or perspective of the system. It is the process of representing a system using various graphical notations that shows how users will interact with the system and how certain parts of the system function. The proposed system was modeled using the following diagrams:

1. Architecture diagram
2. Use case diagram
3. Flowcharts

The proposed system will be implemented using Python Programming language along with different machine learning models and libraries such as pandas, scikit-learn, python who-is, beautiful-Soup, NumPy, seaborn, and matplotlib. Etc.

### 3.5.1 System Architecture

Architectural design is concerned with understanding how a system should be organized and designing the overall structure of that system, it shows how different components of the system work together to achieve its main objectives. It is the process for identifying sub-systems making up a system and the framework for sub-system control and communication. The diagram below represents a graphical overview of the architectural design of the proposed system. Figure 3.1 shows the architecture view of the proposed phishing detection system such that a user enters a URL link and the link moves through different trained machine learning and deep neural network models and the best model with the highest accuracy is selected. Thus, the selected model is deployed as an API (Application Programming Interface) which is then integrated into a web application. Hence, a user interacts with the web application which is accessible across different display devices such as computers, tablets, and mobile devices.

### 3.5.2 Use a case diagram of the system

The Use Case diagram describes the functionality of the system as designed from the requirements; it summarizes the details of a system and the users within the system. It is a behavior diagram and visualizes the observable interactions between actors and the system under development. The Use case diagram consists of the system, the related use cases, and actors and relates to each other.

Figure 3.2 shows the Use case scenarios that a user can carry out on the phishing detection system.

A diagram of a software development process

AI-generated content may be incorrect.

*Figure 3.2: Architectural Design of the Proposed System*

A diagram of a person with text

AI-generated content may be incorrect.

*Figure 3.3 Use Case diagram for Proposed System*

## 3.5.3 Flowchart of the system

A flowchart is a diagram that depicts a process, system, or computer algorithm. It is a graphical representation of the steps that are to be performed in a system, it shows the steps in sequential order. It is used in presenting the flow of algorithms and to communicate complex processes in clear, easy-to-understand diagrams.

Figure 3.3 shows the flow of phishing detection systems using the machine learning process.

Figure 3.4 shows the phishing detection web interface system. The user inputs a URL link and the website validates the format of the URL and then predicts if the link is phishing or legitimate.

A diagram of a process

AI-generated content may be incorrect.

*Figure 3.4 Flowchart of the proposed System*

A diagram of a computer program

AI-generated content may be incorrect.

*Figure 3.5 Flowchart of the web interface*

# **CHAPTER 4: SYSTEM IMPLEMENTATION AND RESULTS**

## **4.1 Software Process Model**

A software process model is an abstraction of the software development process. The models specify the stages and order of a process. So, think of this as a representation of the order of activities of the process and the sequence in which they are performed. There are various types of Software Process Model but in this project, we are going to use Iterative and Increment Model.

**Iterative Incremental Model:**

Since, our project has multiple parts (Agent, API server, ML engine, Dashboard), which can be built and tested in small increments with the help of Iterative Increment Model. It has the following phases:

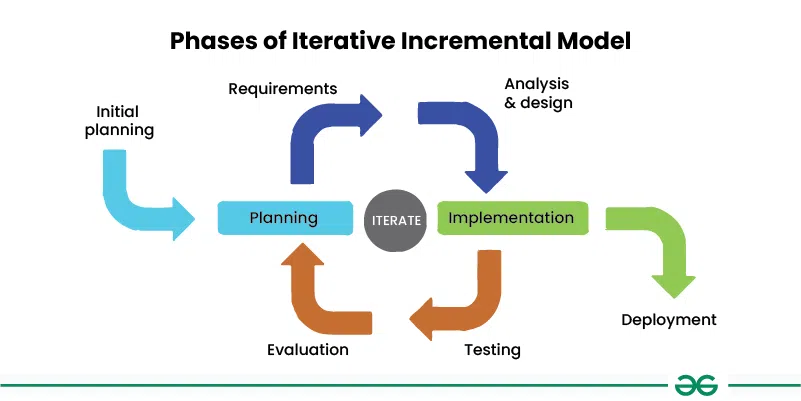


Figure 1 :Phases of Iterative Increment Model

**1. Planning Phase**

In this phase, the team identifies the goals and objectives of the project, along with the project scope, requirements, and constraints on them. The team then identifies different iterations that would be needed to complete the project successfully.

**2. Requirements Analysis and Design Phase**

In this phase, the requirements met are then analyzed and the according system is designed based on these requirements. The projected design should be modular, which would allow easy modification and testing in subsequent iterations.

**3. Implementation Phase**

In this phase, the system is implemented based on the design created in the previous phase. The implementation should be done in small, manageable pieces or increments, which can then be tested in the next phase of the cycle.

**4. Testing Phase**

In this phase, the system is tested against the requirements identified in the planning phase. Testing is done for each iteration, and any defects or issues are identified and resolved, and this helps in each iteration.

**5. Evaluation Phase**

In this phase, the team evaluates the performance of the system based on the results of testing. Feedback is gathered from users and stakeholders, and changes are made to the system as needed, which makes the system more scalable and flexible.

**6. Incremental Release**

In this phase, the completed iterations are released to users and stakeholders. Each release builds on the previous release, providing new functionality or largely improving existing functionality.

Overall, following a structured methodology ensures that the **Phishing URL detection using ML** is developed efficiently and effectively, meets the project requirements, and provides positive user experience for each generation of the user.

# 8. OUTPUT: -

A computer screen shot of a person holding a lock

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.A screenshot of a graph

AI-generated content may be incorrect.

# **9. CONCLUSION: -**

Our project introduces a proactive and intelligent approach to cybersecurity through the development of a lightweight agent-based threat detection system. By continuously collecting system and user activity logs and analyzing them using machine learning techniques, the system can detect anomalies in real-time, allowing for quicker threat identification and response. This solution reduces reliance on manual monitoring and provides organizations with a scalable, cost-effective way to safeguard their digital infrastructure. With its modular design and automated alert system, our project demonstrates a significant step towards modernizing endpoint security and addressing evolving cyber threats.

The contribution of this research work is to help both individuals and organizations identify and understand phishing techniques used by phisher as well as help them detect phishing URL attack effectively and efficiently by employing machine learning models instead of the previous method of detecting phishing website.

# **10. BIBLIOGRAPHY: -**

[1] N. Abdelhamid, F. Thabtah, and H. Abdel-Jaber, "Phishing detection: A recent intelligent machine learning comparison based on models’ content and features," 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), Beijing, China, pp. 72–77, 2017, doi: 10.1109/ISI.2017.8004877.

[2] N. S. Anjum, M. S. Antesar, and M. A. Hossain, "A Literature Review on Phishing Crime, Prevention Review and Investigation of Gaps," 10th Int. Conf. on Software, Knowledge, Information Management & Applications (SKIMA), Chengdu, China, pp. 9–15, 2016, doi: 10.1109/SKIMA.2016.7916190.

[3] A. Almomani, B. B. Gupta, S. Atawneh, A. Meulenberg, and E. Almomani, "A survey of phishing email filtering techniques," IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2070–2090, 2013.

[4] R. M. Rishikesh and S. Irfan, "Phishing Website Detection using Machine Learning Algorithms," International Journal of Computer Applications, vol. 23, pp. 45–47, 2018, doi: 10.5120/ijca2018918026.

[5] R. Kiruthiga and D. Akila, "Phishing Websites Detection Using Machine Learning," ResearchGate, 2019. [Online]. Available: <https://www.researchgate.net/publication/337049054>

[6] B. B. Gupta, A. Tewari, A. K. Jain, and D. P. Agrawal, "Fighting against phishing attacks: State of the art and future challenges," Neural Computing and Applications, 2016.

[7] T. Peng, I. Harris, and I. Sawa, "Detecting Phishing Attacks Using Natural Language Processing and Machine Learning," 12th IEEE International Conference on Semantic Computing (ICSC), pp. 300–301, 2018.

[8] Anti-Phishing Working Group (APWG), Phishing Activity Trends Report – 1st Quarter 2014, [Online]. Available: http://docs.apwg.org/reports/apwg\_trends\_report\_q1\_2014.pdf

[9] Imperva, "Phishing Attacks," 2021. [Online]. Available: <https://www.imperva.com/learn/application-security/phishing-attack-scam/>