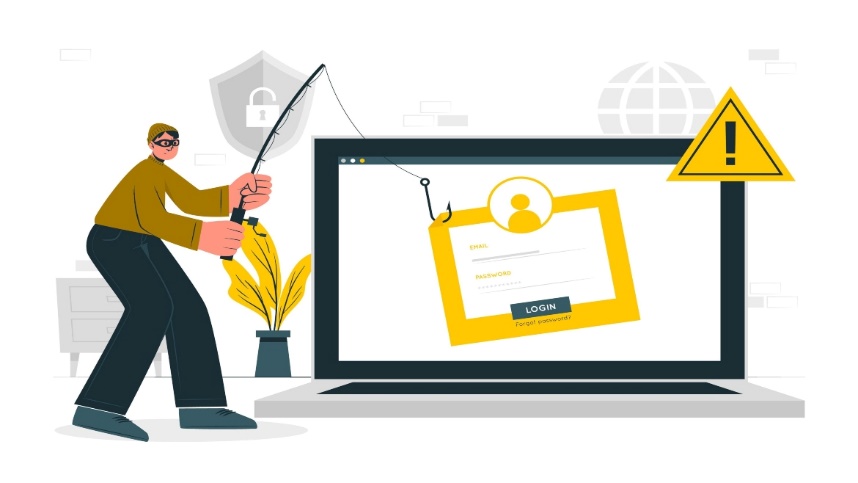
Low Level Document Phishing Domain Detection



Mukhesh Kuna

**Content**

[ABSTRACT… 3](#_TOC_250010)

1. [Introduction… 4](#_TOC_250009)
   1. What is Low Level Design Document 4
   2. [Scope 4](#_TOC_250008)
   3. [Introduction 4](#_TOC_250007)
   4. [Constraints 5](#_TOC_250006)
   5. [Out of Scope 5](#_TOC_250005)
2. [Problem Statement 5](#_TOC_250004)
3. [Dataset Information… 5](#_TOC_250003)
4. [Architecture 12](#_TOC_250002)
   1. [Architecture Description 12](#_TOC_250001)
5. [Logging 12](#_TOC_250000)

# Abstract

Phishing is a fraudulent process where attackers aim to obtain sensitive information, often through emails, text messages, or fake websites. The rise of phishing websites that mimic legitimate ones has drawn attention from the machine learning community. This paper presents a dataset of 58,645 websites labeled as legitimate or phishing, enabling the development of effective phishing detection systems.

# Introduction

* 1. **Why this Low-Level Design Document?**

The primary purpose of this Low-Level Design (LLD) Document is to augment the existing project description with essential details, providing a solid foundation for the coding phase. This document serves as a proactive means of identifying potential contradictions before the coding process commences and serves as a reference manual for high-level module interactions.

The central goal of this project is to predict whether a website is legitimate or potentially malicious. This prediction serves as a critical foundation for taking appropriate actions against malicious or fake websites, contributing to the prevention of phishing attacks.

The project can be effectively delivered in three distinct phases:

* Building Machine Learning Model: This phase involves the development of a machine learning model tailored to the project's specific requirements.
* Integration of UI and Database: The next phase focuses on seamlessly integrating the user interface (UI) and the database with all project functionalities, ensuring a cohesive and efficient user experience.
* Deployment on Local Server: The final phase entails deploying the project on a local server, making it accessible and functional for users.

This structured approach ensures the project's successful development and deployment, with a clear focus on achieving the primary objective of predicting website authenticity and taking action against potential threats.

## Scope

This software system is a web application designed to predict whether a website is genuine or potentially malicious. It leverages user input across several categories, including the Length of the URL, Length of the Directory, Length of the File, Activation Domain Time, and Expiration Domain Time.

The system utilizes these input features to run a predictive model, determining whether the website in question is real or fake. To ensure optimal performance and maximize the company's profits, it is essential that all the required features are available during the prediction process.

This approach guarantees the efficient utilization of data and resources, contributing to better decision-making and enhanced security in the digital landscape.

## Introduction

Phishing is a deceptive form of fraud in which the attacker seeks to obtain sensitive information, including login credentials and account details, by impersonating a trusted entity or individual through email or other communication channels.

Typically, victims receive messages that appear to originate from familiar contacts or reputable organizations. These messages often contain malicious software that targets the recipient's computer or includes links designed to direct victims to counterfeit websites. The aim is to deceive individuals into disclosing personal and financial data, such as passwords, account IDs, or credit card information.

Phishing is a favored tactic among attackers because it is often easier to manipulate someone into clicking on a seemingly legitimate yet malicious link than to bypass a computer's robust defense systems. These deceitful links within the message are skillfully crafted to mimic the appearance of the spoofed organization, complete with the organization's logos and other genuine content, making them particularly convincing and dangerous.

## Constraints

We will be using only few features (Selection of features using Feature Selection method).This system only predict weather the website is real or fake.

## Out of Scope

System will not perform well, if user input is not a valid URL.

1. **Problem Statement**

Phishing is a type of fraud in which an attacker impersonates a reputable company or person through email or other communication channels to obtain sensitive information such as login credentials or account details. Attackers prefer phishing because convincing someone to click a seemingly authentic malicious link is easier than bypassing a computer's security measures. The primary objective is to predict the authenticity of domains, distinguishing between genuine and malicious ones.

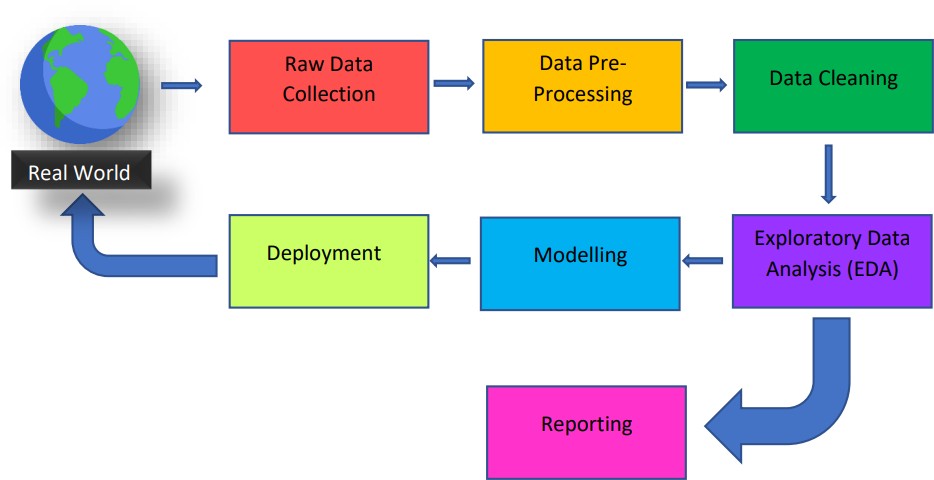
1. **Dataset Information**
   1. **Dataset overview**

Dataset consist of 58654 records and 112 features. This dataset mainly divided into four parts:

* + 1. URL-Based Features
    2. Domain-Based Features
    3. Page-Based Features
    4. Content-Based Features

The presented dataset was collected and prepared for the purpose of building and evaluating various classificatio n methods for the task of detecting phishing websites based on the uniform resource locator (URL) properties, U RL resolving metrics, and external services.

1. **Architecture**



1. **Logging**

In our system, it is imperative to maintain a comprehensive log of events to keep users informed about the internal processes that are running. The initial steps for achieving this are as follows:

Identification of Logging Steps: The system intelligently identifies the specific steps at which logging is required to ensure that it captures crucial events.

Logging of System Flow: Our system is designed to log each and every step of the system's flow. This comprehensive approach ensures that no critical event or action goes unrecorded.

Logging Method Selection: As a part of system flexibility, developers have the autonomy to select their preferred logging method. For our system, we have chosen to implement File logging, which offers an effective means of capturing and storing log data.

Maintaining System Performance: Importantly, our system is designed in a way that ensures it does not hang or experience performance degradation due to the use of file logging. Logging is not just an optional feature; it is a mandatory component of our system.