High Level Design (HLD)

PHISHING DOMAIN DETECTION

Revision Number – 1.3

Last Date of Revision – 04-feb-2023

Abhishek

Document Version Control:

Date	Version	Author	Description
27-01-2023	1.0	Abstract, Introduction Problem Statement	Abhishek
30-01-2023	1.1	Design Flow	Abhishek
04-feb-2023	1.2	Performance Evaluation	Abhishek

Contents

Abstract

Phishing is a type of fraud in which an attacker impersonates a reputable company or person in order to get sensitive information such as login credentials or account information via email or other communication channels. Phishing is popular among attackers because it is easier to persuade someone to click a malicious link that appears to be authentic than it is to break through a computer's protection measures.

1. Introduction

1.1. What is High-Level design document?

The main purpose of this HLD documentation is to feature the required details of the project and supply the outline of the machine learning model and also the written code. This additionally provides the careful description on however the complete project has been designed end-to-end.

1.2. Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

2. Description

2.1. Problem Perspective

Phishing is a type of fraud in which an attacker impersonates a reputable company or person in order to get sensitive information such as login credentials or account information via email or other communication channels.

2.2. Problem Statement

The mail goal is to predict whether the domains are real or malicious.

2.3. Purposed Solution

Projected to require the desired input of user from the created interface and method all the provided information to satisfy the wants of the machine learning model and at last show the output oral communication so and then quantity is that the expected value.

2.4. Solution Improvements

We will even predict the phishing tag considering whether by getting the mails a user receives on weeks , promotional messages and the malicious links that user has clicked.

Source of mails ,messages etc. should be take care of to improve the solution.

2.5. Technical Requirements

There are not any hardware needs needed for victimization this application, the user should have AN interactive device that has access to the web and should have the fundamental understanding of providing the input. And for the backend half the server should run all the package that's needed for the process the provided information and to show the results.

2.6. Data Requirements

The info demand is totally supported the matter statement. And also, the information set is accessible on the Kaggle within the type of standout sheet(.xlsx). Because the main theme of the project is to induce the expertise of real time issues, we have a tendency to once more mercantilism {the information into the prophetess data base and commerce it into csv format.}

2.7. Tool Used

- Python 3.9 is employed because the programming language and frame works like
- Numpy, pandas, sklearn and alternative modules for building the model.
- VsCode is employed as IDE.
- For visualizations seaborn and components of matplotlib and seaborn are getting used.
- GitHub is employed for version management.
- For Deployment AWS is used
- Dockers and Guthub Actions is used for CI CD piplelining
- Apache Airlflow is being used for model monitoring

2.8. Constraints

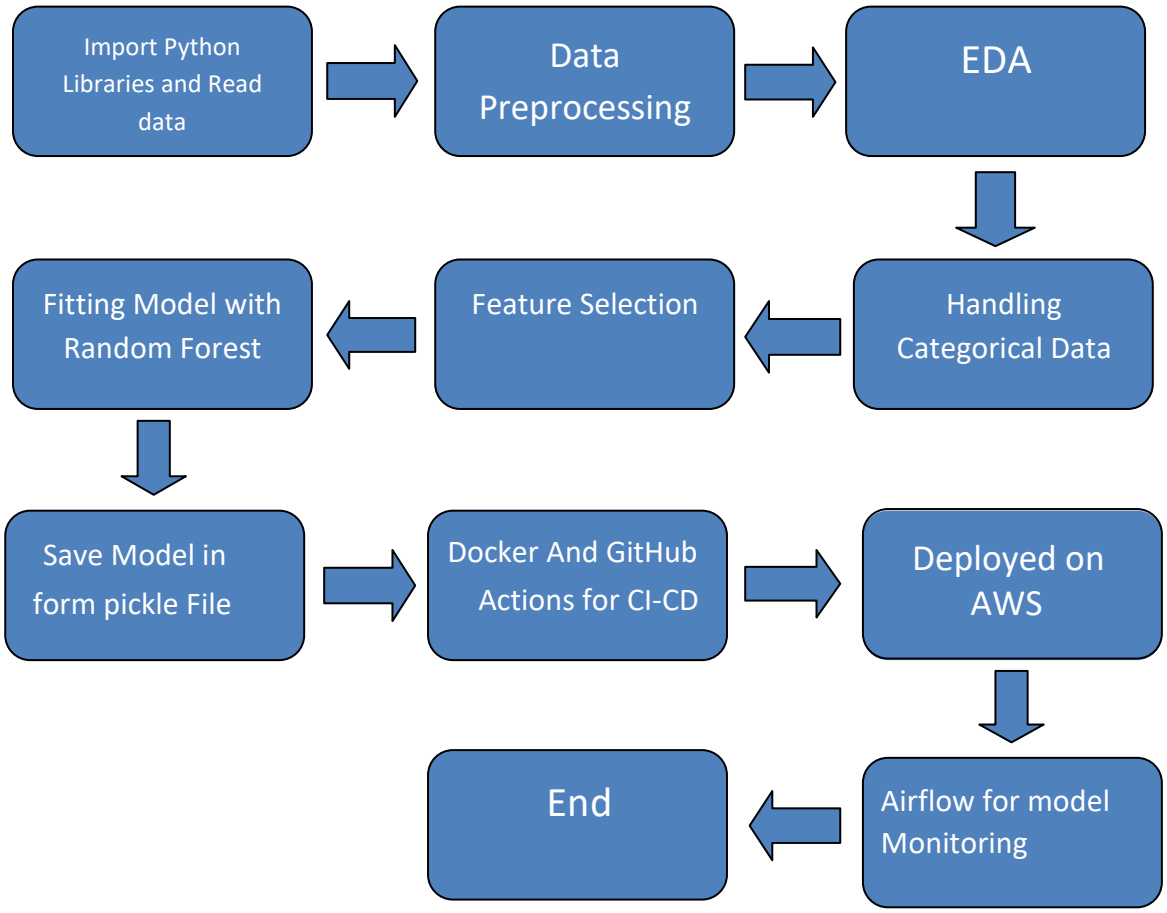
The Phishing Domain Detection answer should be understandable by industry , as automatic as attainable and also the user should not be needed to understand any of the operating.

2.9. Assumptions

The most objective of the project is to implement the utilization cases as for the new dataset that user provides through the programme. Machine learning model is employed for process the on top of computer file. It's additionally assumed that each one aspects of this project have the flexibility to figure along within the approach the designer is expecting.

3. Design Flow

3.1. Modeling Process



3.2. Logging

Each step is being logged within the system that runs internally, that shows the date time and therefore the processed that has been performed, work is completed in several layers as information, DEBUG, ERROR, and WARNINGS. This provides us the perceive of the logged info

3.3. Error Handling

Once ever a slip is occurred, the reason are logged in its several log file, in order that the developer will rectify the error

4. Performance Evaluation

4.1. Reusability

Elements of the code written is accustomed different applications and therefore the rest is changed and be reused.

4.2. Application Compatibility

The various parts for this project are exploitation python as associate interface between them. Every element can have its own tasks to perform, and it's the work of the python to make sure correct transfer of data.

4.3. Resource Utilization

Once any task is performed, it'll doubtless use all the process power offered till that perform is finished.

5. Conclusion

The Phishing Domain Detection can be triggered on Apache Airflow to predict whether Transaction is fraud or not