Phishing Domain Detection

Objective:

Development of a classifier model for “ Phishing Domain Detection”. The model will determine whether the URL is phishing or non-phishing.

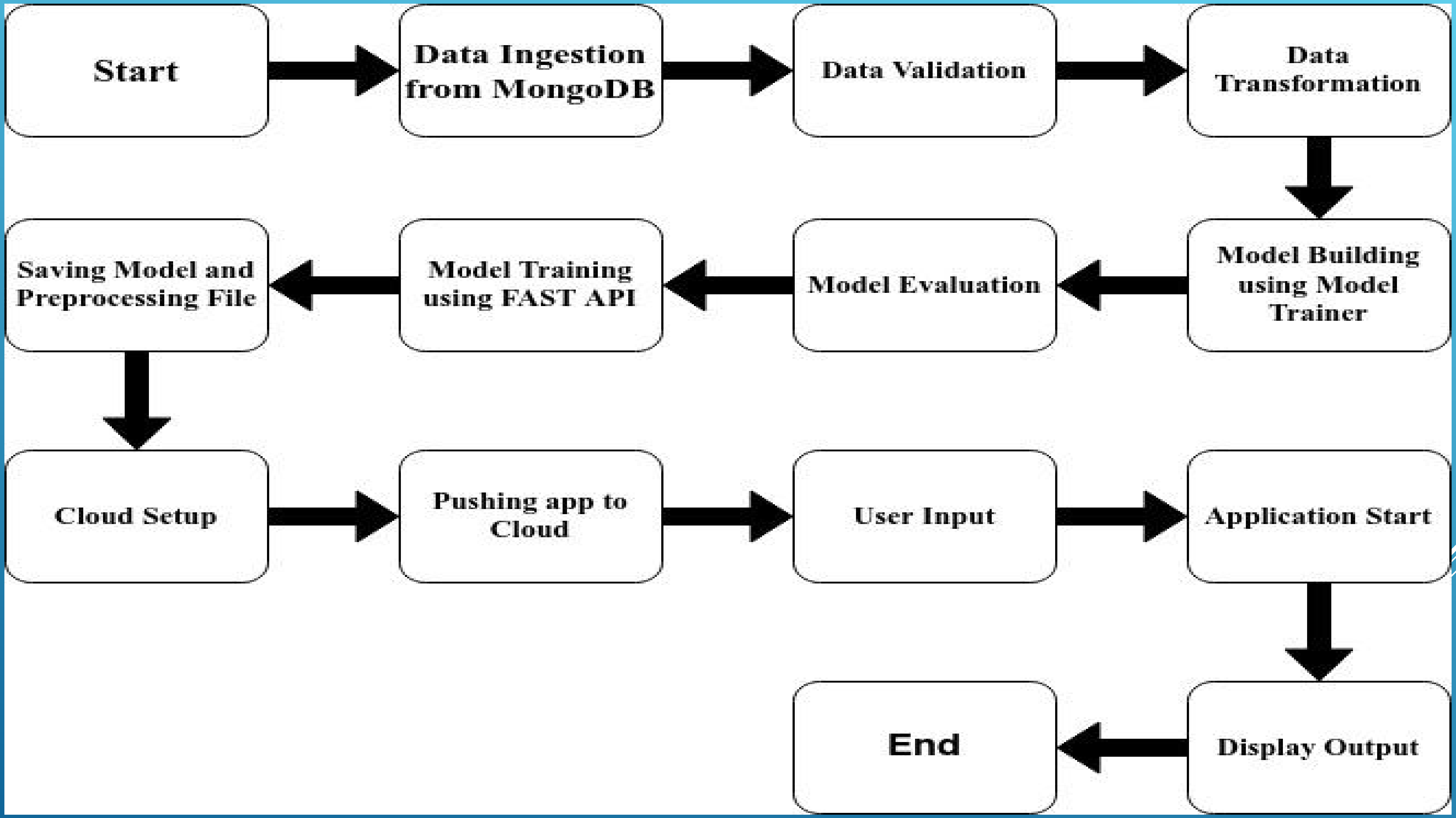
Benefits:

- ✓ Safe Surf
- ✓ Maintence Privacy
- ✓ Helps user to figure out Website URL is Legitimate or Phishing.
- ✓ Saving from Tricksters .

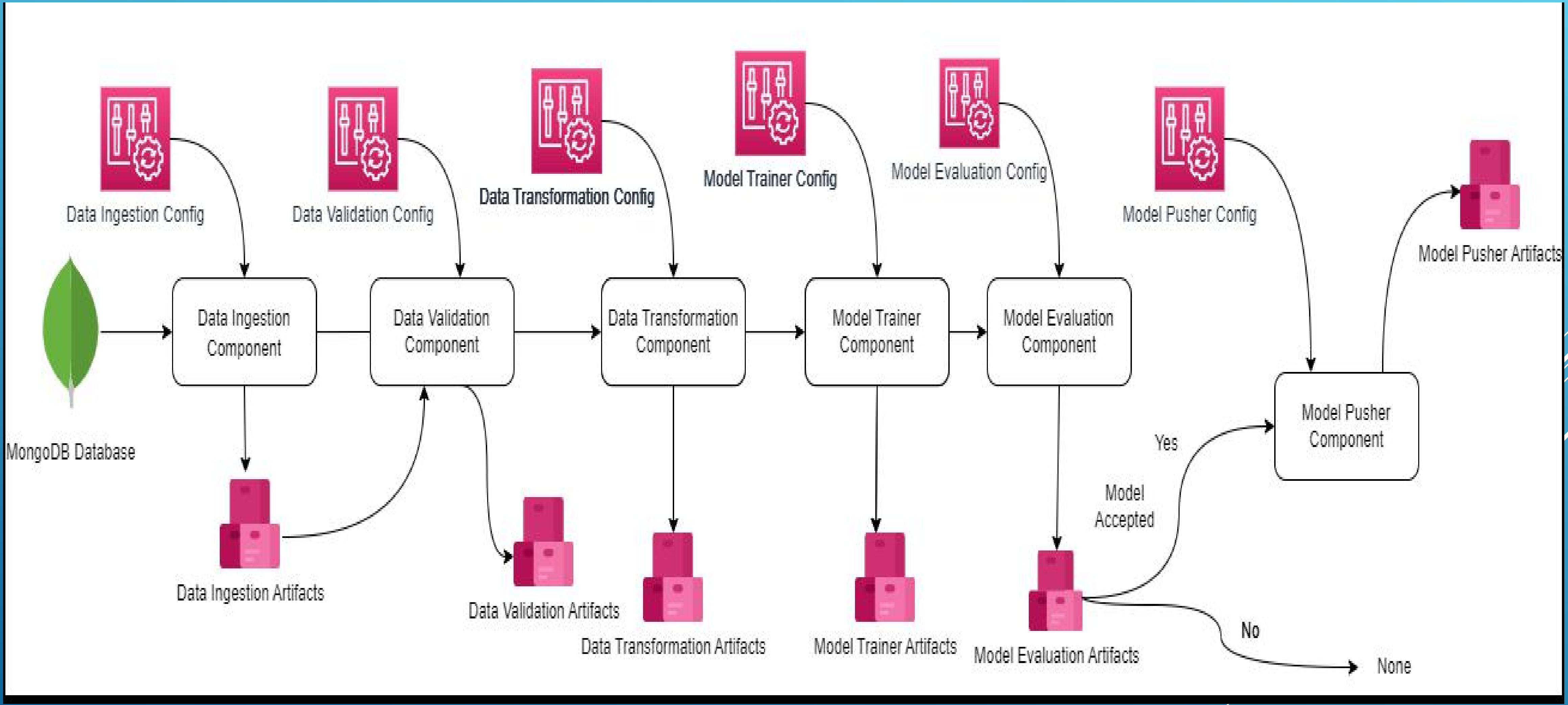
Data Sharing Agreement :

- Sample file name
- Number of Columns
- Column names
- Column data type

Architecture - I



Architecture - II



Data Ingestion:

Data Insertion into Database:

a. Database creation and connection - create a database with name passed. If the database is already created, open the connection to the database.

b. Table creation in the database.

c. Insertion of files in the table

Data Insertion into Database:

- ✓ Data export from database - The data in a stored database is exported as a .CSV file to be used for data pre-processing and model Training.

Data Validation:

Number of Columns – Validation of number of columns present in the files, and if it doesn't match then the file is moved to "Bad_Data_Folder.“

Name of Columns - The name of the columns is validated and should be the same as given in the schema file. If not, then the file is moved to "Bad_Data_Folder".

Data type of columns - The data type of columns is given in the schema file. It is validated when we insert the files into Database. If the datatype is wrong, then the file is moved to "Bad_Data_Folder".

Null values in columns - If any of the columns in a file have all the values as NULL or missing, we discard such a file and move it to "Bad_Data_Folder".

Data Transformation:

- ✓ As soon as data is successfully validated from data validation stage, The validated data is then sent to the data transformation stage where data is wrangled, cleaned by using some data imputation techniques.
- ✓ Furthermore if outliers are present then apply some outliers handling techniques. Next, if dataset is not having proper scaling then will apply proper scaling techniques to trasnform tha data. Now data is ready to next stage i.e. Model Trainer.

Model Trainer:

- ✓ In this stage, Number of classification type supervised machine learning models should be apply and will try to find out the best suitable model as per our availbale dataset by using some metric evaluation techniques.
- ✓ After getting the best suitable classification model, its traning and testing ccuracy will be determined again.

Model Evaluation:

- ✓ In this stage, selected model is evaluated using some criteria like training and testing accuracy with the help of proper metric evaluation techniques, overfitting and underfitting etc.

- ✓ once this selected model is properly evaluated, it will be passed to the pipeline stage.

Model Training using FAST API:

- ✓ After successful run of individual component, a pipeline will be created where each and every component discussed above will merge and that pipeline will be triggered using FAST API where the finalised model shall be trained with availabe training dataset.

Saving Model and Preprocessing File:

- ✓ In this stage, the trained model file plus the preprocessing file will be stored in pickeled format.

Cloud Setup and Model Pushing:

- ✓ After the model/product/app is ready, the app is deplyed to AWS by using some AWS services like Elastic Beanstalk (EB) and Docker hub.
- ✓ This cloud deplymnet will help user to access the application through any internet devices.

User Input and Application output:

- ✓ By using the app URL, user will get access to the application/model, where user will feed inputs to the model and that model will render the best outcome to its webpage in terms of givel URL is Legitimate or Phishing.

Conclusion:

The classification type supervised machine learning model takes URL as input and predicts whether the given URL is Legitimate or Phishing one.

Questions and Answers:

Q-1: What's the source of data?

- ✓ This data set we will be using is from the Mendeley Phishing Websites Dataset. The following is Mendeley data set:
- ✓ Source:

  Published: 24 September 2020| Version 1 | DOI: 10.17632/72ptz43s9v.1
  Contributor: Grega Vrbančič.

- ✓ Dataset link: https://data.mendeley.com/datasets/72ptz43s9v/1 (csv format)

Q-2: What was the type of data?

- ✓ The data was the combination of numerical and boolean values.

Q-3: After the File validation what you do with incompatible file or files which didn’t pass the validation?

- ✓ Files like these are moved to the Achieve Folder and a list of these files has been shared with the client and we removed the bad data folder.

Q-4: How logs are managed?

- ✓ Use of different logs as per the steps that will follow in validation and modelling component like File validation log, Data Insertion log, Model Training log, prediction log etc.

Q-5: What techniques were you using for data pre-processing ?

- ✓ Removing unwanted attributes
- ✓ Visualizing relation of independent variables with each other and output variables
- ✓ Checking and changing Distribution of continuous values
- ✓ Removing outliers
- ✓ Cleaning data and imputing if null values are present.
- ✓ Converting categorical data into numeric values (if present).

Q-6: How training was done or what models were used?

- ✓ Before dividing the data in training and validation set, pre-processing is performed over the data set and made the final dataset or transformed dataset.
- ✓ As per the dataset training and testing data were divided.
- ✓ Classifier algorithms like Logistic regression, SVM, Decision Tree, Random Forest, XGBoost were used based on the Accuracy, Precision, Recall, F1-Score and Final model was used to train on the dataset and saved in pickeled format.

Q-7: How Prediction was done?

- ✓ The testing files are shared by the client.
- ✓ Afterwards, on the basis of dataset, model is selected and prediction is performed.

Q-8: What are the different stages of deployment?

- ✓ First, the scripts are stored on GitHub as a storage interface.
- ✓ The model is first tested in the local environment.
- ✓ After successful testing, it is deployed on AWS via Docker hub.

Thank You !!