ARCHITECTURE DOCUMENT DESIGN (ADD)



**Submitted by:**

Suchindra Kumar

**Document Control: Change Record:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
| 0.1 | 15-june- 2023 | Suchindra Kumar | Initial Architecture design Draft. |
| 0.2 | 15-june-2023 | Suchindra Kumar | Architecture Design appended and updated |
| 0.3 | 16-june-2023 | Suchindra Kumar | Final design appended |

**Reviews:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Reviewer** | **Comments** |
| 1.0 | 18-july-2023 |  | Document Content, Architecture Design Updated |

**Approval Status:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Reviewed By** | **Approved By** | **Comments** |
| 1.1 | 25-july-2023 | Suchindra Kumar |  | Final |

**Contents**

1. [Introduction 4](#_TOC_250014)
   1. What is Architecture Document Design? 4

[1 2. Scope 5](#_TOC_250013)

1. Credit card default prediction Home Page 6
2. [Architecture Description 7](#_TOC_250012)
   1. [Data Description 7](#_TOC_250011)
   2. [Data Transformation 7](#_TOC_250010)
   3. [Exploratory Data Analysis 7](#_TOC_250009)
   4. [Data Cleaning 7](#_TOC_250008)
   5. [Data Preprocessing 8](#_TOC_250007)
   6. [Feature Engineering 8](#_TOC_250006)
   7. [Feature Selection 8](#_TOC_250005)
   8. Model Building 8
   9. Random Forest 8
   10. [Decision Tree Classifier 8](#_TOC_250004)
   11. [Logistic Regression 9](#_TOC_250003)
   12. Support Vector classifier 9
   13. [KNN classifier 9](#_TOC_250002)
   14. [Model Validation 9](#_TOC_250001)
3. [Database 9](#_TOC_250000)
4. Unit Test Case 10

## Introduction:

* 1. **What is an Architecture design document?**

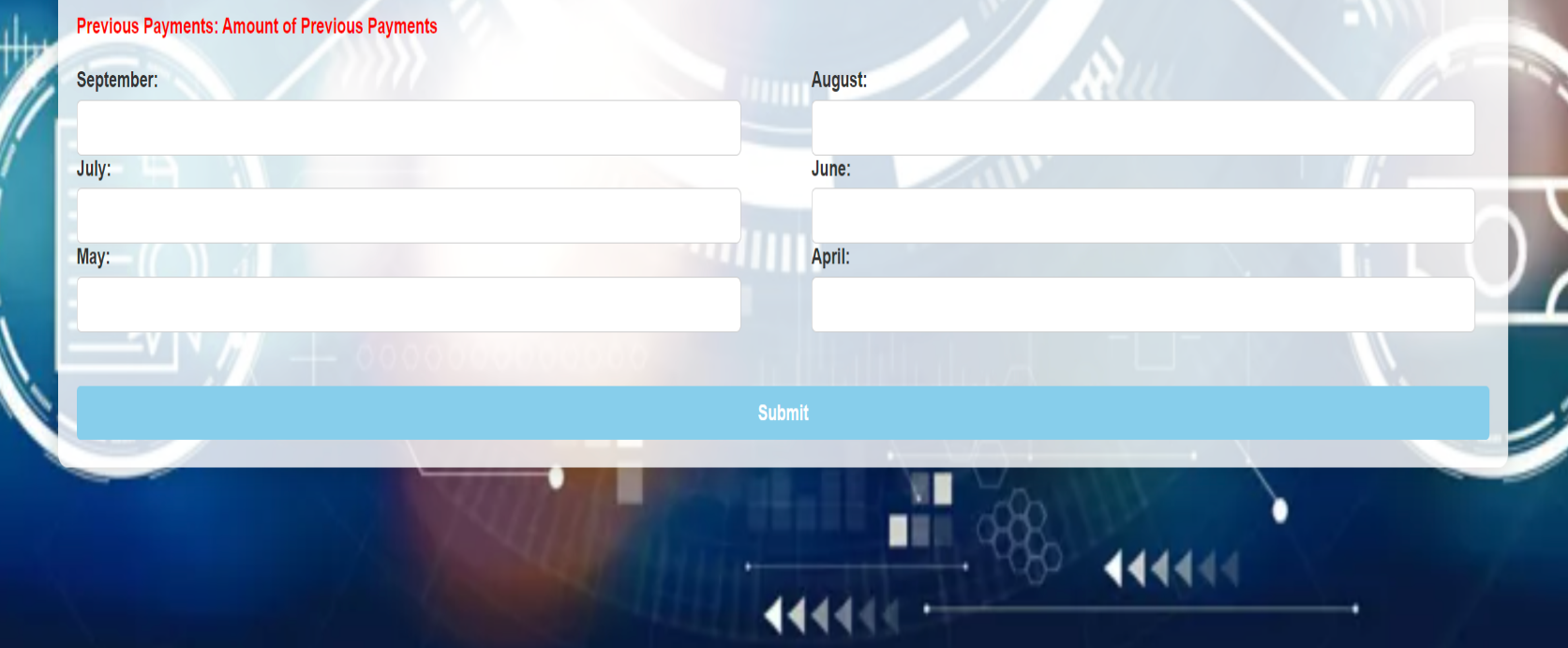
The architecture design document is a technical document describing the components and specifications required to support the solution and ensure that the specific business and technical requirements of the design are satisfied. The architecture design document includes the conceptual, logical, and physical designs.

## Scope:

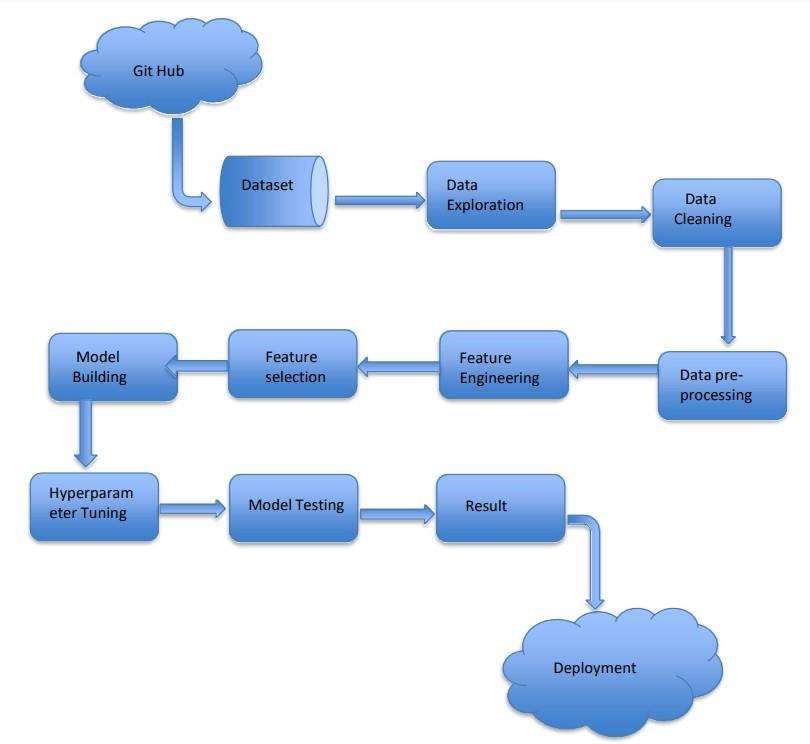
**T**he scope of this document is specific to the implementation of the Predictive Machine Learning Model to deploy on to the cloud. The supporting Libraries are defined in the Architecture Design. These python libraries are used to mining the data to understand the anomalies and frequent patterns from the row data. The resultant data can be used for predicting future sales volume with the help of different machine learning techniques.

**1.3 Credit Card Default Prediction Form Page:**

****



1. **Architecture:**



# Architecture Description:

## Data Description:

The dataset of this research has been published in Kaggle Data Source: <https://www.kaggle.com/datasets/uciml/default-of-credit-card-clients-dataset>

The dataset contains 25 attributes with 30,000 observations for the training and testing sets, respectively.

## Data Transformation:

In the Transformation Process, we will convert our original dataset which is in CSV Format to a Pandas Data Frame. After reading the dataset we have two Pandas data frame train and test concatenated into one data frame.

## Exploratory Data Analysis:

Exploratory Data Analysis, or EDA, is an important step in any Data Analysis or Data Science project EDA. A is the process of investigating the dataset to discover patterns, and anomalies (outliers), and form hypotheses based on our understanding of the dataset.

## Data Cleaning:

Data cleansing or data cleaning is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data.

## Data Pre-processing:

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. When creating a machine learning project, it is not always the case that we come across clean and formatted data.

## Feature Engineering:

Feature engineering is the process of using domain knowledge of the data to create features that make machine learning algorithms work. If feature engineering is done correctly, it increases the predictive power of machine learning algorithms by creating features from raw data that help facilitate the machine learning process.

## Feature Selection:

A feature selection algorithm can be seen as the combination of a search technique for proposing new feature subsets, along with an evaluation measure which scores the different feature subsets. The simplest algorithm is to test each possible subset of features finding the one which minimizes the error rate. This is an exhaustive search of the space, and is computationally intractable for all but the smallest of feature sets.

* 1. **Model Building:**

A machine learning model is built by learning and generalizing from training data, then applying that acquired knowledge to new data it has never seen before to make predictions and fulfil its purpose. Lack of data will prevent you from building the model, and access to data isn't enough.

* 1. **Random Forest:**

A random forest classifier, a random forest is a Meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

## Decision Tree Classifier:

Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

## Logistic Regression:

Logistic regression is a statistical model used for binary classification. It predicts the probability of an event belonging to one of two classes. It uses the logistic function (sigmoid function) to transform the input variables into a value between 0 and 1, representing the estimated probability. The model provides interpretable coefficients for each input variable, indicating their influence on the predicted probability. Logistic regression is trained using maximum likelihood estimation and can be evaluated using metrics such as accuracy, precision, recall, and the ROC curve.

* 1. **Support Vector classifier:**

The Support Vector Classifier (SVC), also known as Support Vector Machine (SVM), is a classification algorithm. It finds an optimal hyperplane that separates data points of different classes with the maximum margin. SVC can handle nonlinear data by using the kernel trick, and it aims to find a hyperplane with the largest margin between support vectors. It is trained by solving an optimization problem, and it can classify new data points based on their position relative to the hyperplane.

## KNN Classifier:

KNN is a non-parametric algorithm that classifies data points based on their proximity to other data points. It assigns a class label to an unclassified data point based on the class labels of its nearest neighbors in the feature space. The "K" in KNN represents the number of nearest neighbors considered for classification. When a new data point is to be classified, KNN finds the K closest neighbors based on a distance metric (such as Euclidean distance) and assigns the class label that appears most frequently among those neighbors.

## 3.14. XGBoost Classifier:

XGBoost (eXtreme Gradient Boosting) is a powerful, efficient machine learning algorithm based on gradient boosting, widely used for classification and regression tasks. It excels in performance due to features like regularization, parallel processing, tree pruning, and automatic handling of missing values

## 3.15. Model Validation:

For machine learning systems, we should be running model evaluation and model tests in parallel. Model evaluation covers metrics and plots which summarize performance on a validation or test dataset. Model testing involves explicit checks for behaviors that we expect our model to follow.

## 3.16 Flask:

Flask is a micro web framework written in python. Here we use flask to create an API that allows us to send data, and receive a prediction as a response. Flask supports extensions that can add application features as if they were implemented in Flask itself.

## Database:

MongoDB is a popular, open-source, document-oriented NoSQL (Not Only SQL) database program. It is designed to store data in a format similar to JSON (JavaScript Object Notation), making it highly flexible and easy to work with, especially for developers familiar with JSON-style data. MongoDB is known for its scalability, high performance, and ability to handle large volumes of data across distributed systems.

**Unit test Cases:**

|  |  |  |
| --- | --- | --- |
| **Test Case Description** | **Pre-requisite** | **Expected Result** |
| Verify the user should able to see their input data | 1. Application is accessible 2. Application is responsive | Users can see their data in the form. |
| Verify the user can edit their information in the form | 1. Application is responsive | User should be able to edit all input field |
| Verify whether the user gets Submit button to submit the inputs. | 1. Application is accessible   2. Application is responsive | User should able to submit values |
| Verify whether the wrong information should not be submitted by the user | Application is secured | Correct information should be submitted |
|  |  |  |