# Architecture
# Credit Card Default Prediction Project

## *1. Introduction*

The Architecture Document for the Credit Card Default Prediction project provides a detailed overview of the system's architecture and components involved in developing, evaluating, and deploying a predictive model for credit card default prediction.

## *2. Architecture Overview*

The architecture for the Credit Card Default Prediction project is designed to handle the following key tasks:

- Data collection and preprocessing
- Feature engineering
- Model development and training
- Model evaluation and selection
- Model deployment
- Monitoring and maintenance

The architecture follows a typical end-to-end machine learning pipeline, ensuring a systematic and efficient flow of data and processes.

## *3. Architectural Components*

The Credit Card Default Prediction project architecture consists of the following components:

### *3.1. Data Collection*

- Responsible for retrieving credit card transaction data, customer information, and default labels from appropriate sources, such as databases, data warehouses, or APIs.
- Ensures data integrity, accuracy, and privacy compliance during data retrieval.

### *3.2. Data Preprocessing*

- Handles data cleaning by addressing missing values, outliers, and inconsistencies in the dataset.
- Performs data normalization, feature scaling, and transformation to prepare the data for modeling.
- Splits the dataset into training, validation, and testing sets to support model development and evaluation.

### *3.3. Feature Engineering*

- Focuses on creating additional features that can enhance the predictive power of the model.
- Utilizes domain knowledge and data analysis techniques to engineer relevant features.
- Incorporates feature selection methods to identify the most important features for credit card default prediction.

### *3.4. Model Development and Training*

- Selects suitable machine learning algorithms, such as logistic regression, decision trees, random forests, or gradient boosting, based on the project requirements.
- Implements the selected algorithms using appropriate libraries or frameworks, such as scikit-learn.
- Utilizes the training dataset to train and optimize the model parameters.

### *3.5. Model Evaluation and Selection*

- Defines evaluation metrics, such as accuracy, precision, recall, F1-score, and area under the ROC curve (AUC-ROC), to assess the model's performance.
- Evaluates multiple models using validation datasets to compare and select the best-performing model.
- Considers factors like model interpretability, computational efficiency, and business requirements during model selection.

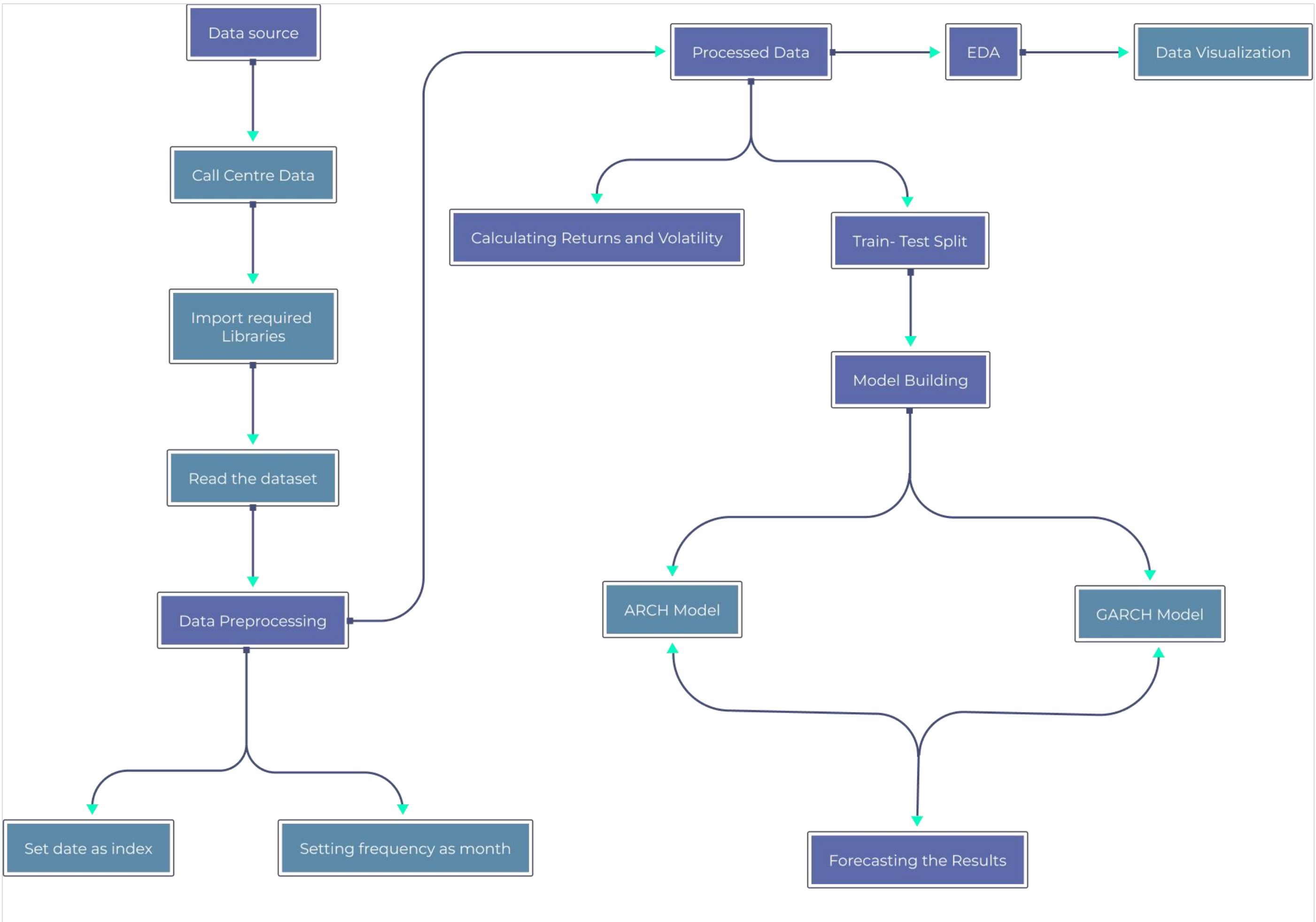
### *3.6. Model Deployment*

- Deploys the selected model into a production environment, making it available for real-time predictions.
- Integrates the model with the existing credit card processing system, ensuring seamless data flow and compatibility.
- Establishes appropriate APIs or endpoints to receive input data and provide predictions.

### *3.7. Monitoring and Maintenance*

- Implements a monitoring system to track the deployed model's performance and behavior in the production environment.
- Monitors prediction accuracy, response time, and other relevant metrics to detect anomalies or performance degradation.
- Conducts regular maintenance activities, including model retraining, feature updates, and bug fixes to ensure optimal performance.

## *4. Architecture Diagram*



## 5. Deployment Considerations

- Considers infrastructure requirements, such as computing resources, storage capacity, and network connectivity, to support the deployment of the predictive model.
- Adheres to security protocols and data privacy regulations during data transmission and storage.
- Implements load balancing and fault tolerance mechanisms to handle varying workloads and ensure high availability.

## 6. Scalability and Performance

- Designs the architecture to handle large-scale datasets and accommodate future growth.
- Utilizes parallel processing or distributed computing techniques to enhance performance and reduce processing time.
- Optimizes the model and system components for efficient resource utilization and scalability.

## 7. Conclusion

The Architecture Document provides a comprehensive overview of the Credit Card Default Prediction project's system architecture. It outlines the key components involved in data collection, preprocessing, feature engineering, model development, evaluation, deployment, monitoring, and maintenance. Following this architecture will enable the successful implementation of the project and the development of an accurate and reliable credit card default prediction model.