# Disclaimer: This output contains AI-generated content; user is advised to review it before consumption.  
  
#\*Start of AI Generated Content\*

# **Synthetic Data Generator API Technical Document**

## **Introduction**

This document provides a comprehensive overview of the Synthetic Data Generator API, a Python-based application utilizing Flask, scikit-learn, and pandas. The API enables users to train machine learning (ML) models and generate synthetic data while ensuring data anonymization. This introduction sets the stage for a deeper dive into the problem statement, solution approach, code functionality, input/output formats, and conclusion.

## **Problem Statement**

* **Limited Data Availability**: Insufficient data for training robust ML models.
* **Data Privacy Concerns**: Sensitive information exposure during data sharing or model training.
* **Model Generalizability**: Need for diverse, representative data to enhance model performance.

## **Solution Approach**

* **Modular Design**: Separate functions for data loading, ML model training, synthetic data generation, and anonymization.
* **Scalable Framework**: Utilize Flask for API development, enabling easy integration and scalability.
* **State-of-the-Art Libraries**: Leverage scikit-learn for ML tasks and pandas for efficient data manipulation.
* **Data Anonymization**: Employ hashing to protect sensitive information.

## **Functionality of Code**

### **Constants and Static Values**

* Defined at the top of the script for clarity and ease of modification:
* APP\_NAME, APP\_VERSION
* DEFAULT\_ML\_ALGORITHM, SUPPORTED\_ML\_ALGORITHMS
* DATA\_SECURITY\_PROTOCOL, ANONYMIZATION\_TECHNIQUE
* MAX\_SYNTHETIC\_DATA\_SIZE, MIN\_SYNTHETIC\_DATA\_SIZE, DEFAULT\_SYNTHETIC\_DATA\_SIZE

### **Import Statements**

* Essential libraries for the application:
* os, logging
* pandas as pd
* sklearn.ensemble for RandomForestClassifier
* sklearn.svm for SVC
* sklearn.model\_selection for train\_test\_split
* sklearn.metrics for accuracy\_score
* numpy as np
* hashlib, hmac, base64 for data anonymization
* Flask for API development

### **Logging Configuration**

* Basic logging setup with level set to INFO

### **Exception Handling**

* Custom exception classes:
* SyntheticDataGeneratorError (base)
* InvalidMLAlgorithmError
* DataSecurityError

### **Functions**

* **\_load\_data(file\_path: str) -> pd.DataFrame**:
* Loads data from a CSV file.
* Handles exceptions, logging errors.
* **\_train\_ml\_model(data: pd.DataFrame, algorithm: str) -> object**:
* Trains an ML model based on the provided data and algorithm.
* Supports "Random Forest" and "Support Vector Machine".
* Evaluates model accuracy.
* **\_generate\_synthetic\_data(model: object, size: int) -> pd.DataFrame**:
* Generates synthetic data using a trained ML model.
* Size parameter controls the amount of data generated.
* **\_anonymize\_data(data: pd.DataFrame) -> pd.DataFrame**:
* Anonymizes data by hashing object-type columns.

### **API Endpoints**

* **/train\_ml\_model (POST)**:
* Request Body: file\_path, algorithm
* Trains an ML model, returns model metadata (ID, algorithm)
* **/generate\_synthetic\_data (POST)**:
* Request Body: model\_id, size
* Generates synthetic data, anonymizes it, returns data metadata (ID, size)

## **Input and Output Format**

### **Input Format**

* **/train\_ml\_model**:
* Content-Type: application/json
* { "file\_path": "path/to/your/data.csv", "algorithm": "Random Forest" }
* **/generate\_synthetic\_data**:
* Content-Type: application/json
* { "model\_id": <model\_id\_returned\_earlier>, "size": 1000 }

### **Output Format**

* **Success Responses**:
* Content-Type: application/json
* { "model\_id": "<model\_id>", "algorithm": "<algorithm>" } (for /train\_ml\_model)
* { "data\_id": "<data\_id>", "size": <size> } (for /generate\_synthetic\_data)
* **Error Responses**:
* Content-Type: application/json
* { "error": "<detailed\_error\_message>" }
* HTTP Status Code: 500 Internal Server Error

## **Conclusion**

The Synthetic Data Generator API effectively addresses the challenges of limited data availability and data privacy concerns by providing a scalable, secure, and easy-to-use solution for training ML models and generating anonymized synthetic data. This technical document has outlined the problem statement, solution approach, code functionality, and input/output formats, offering a comprehensive understanding of the API's capabilities and usage.

#\*End of AI Generated Content\*