Project Summary

Batch details	PGP in Data Science with Specialization in GenAI
Team members	Shreya Bhat Bento M J Fernandes Shristi Agrawal Chinmayee Y Venkatesh Pattar
Domain of Project	Generative AI
Proposed project title	Generative AI for test Data synthesis
Group Number	6
Team Leader	Shreya Bhat
Mentor Name	Vaibhav Kulkarni

Date:

Signature of the Mentor

Signature of the Team Leader

Table of Contents

1. Problem Statement and Key findings

1.1. Problem Statement

Modern data-driven applications require large volumes of high-quality data for model development, testing, and analysis. However, real-world datasets often contain sensitive personal or business-critical information, making direct usage risky due to privacy, compliance, and ethical concerns.

The objective of this project was to design and implement an end-to-end synthetic data generation platform that can:

- Generate realistic synthetic tabular data
- Preserve statistical utility
- Minimize privacy risks
- Provide interpretable evaluation metrics through a user-friendly interface

The platform takes real tabular datasets as input, generates synthetic data using CTGAN, and evaluates the output across utility, statistical similarity, and privacy dimensions.

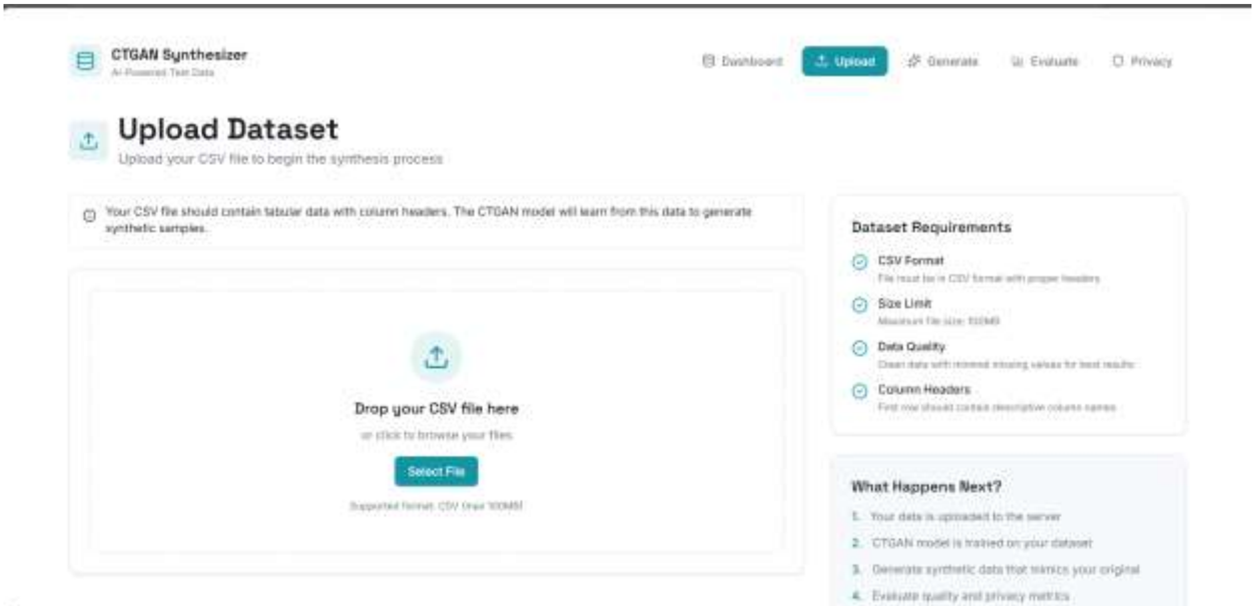
1.2. Key Findings

- CTGAN successfully captured complex relationships between numerical and categorical features.
- Synthetic datasets achieved high utility scores, validated using Train-on-Synthetic-Test-on-Real (TSTR) evaluation.
- Privacy risks such as record disclosure and identifiability remained well below accepted thresholds.
- The system provides transparent, interpretable metrics that help users confidently adopt synthetic data.

2. Overview

The final solution follows a modular, pipeline-based methodology:

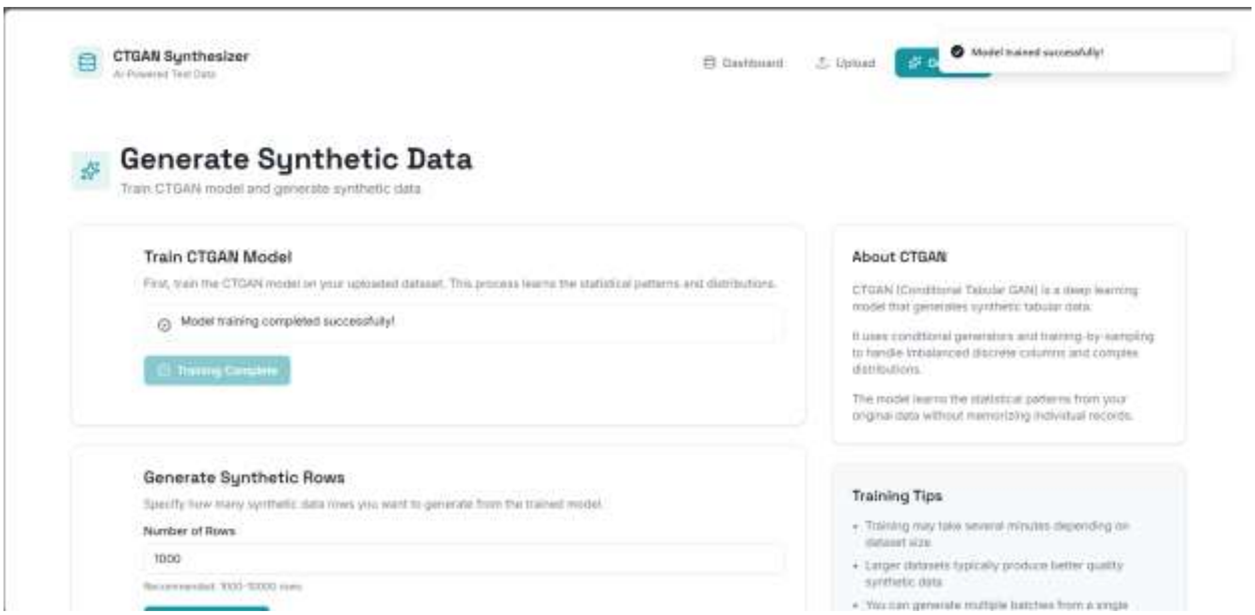
1. **Data Ingestion**
    - User uploads a CSV dataset via the frontend.
    - Dataset is validated and stored securely on the backend.



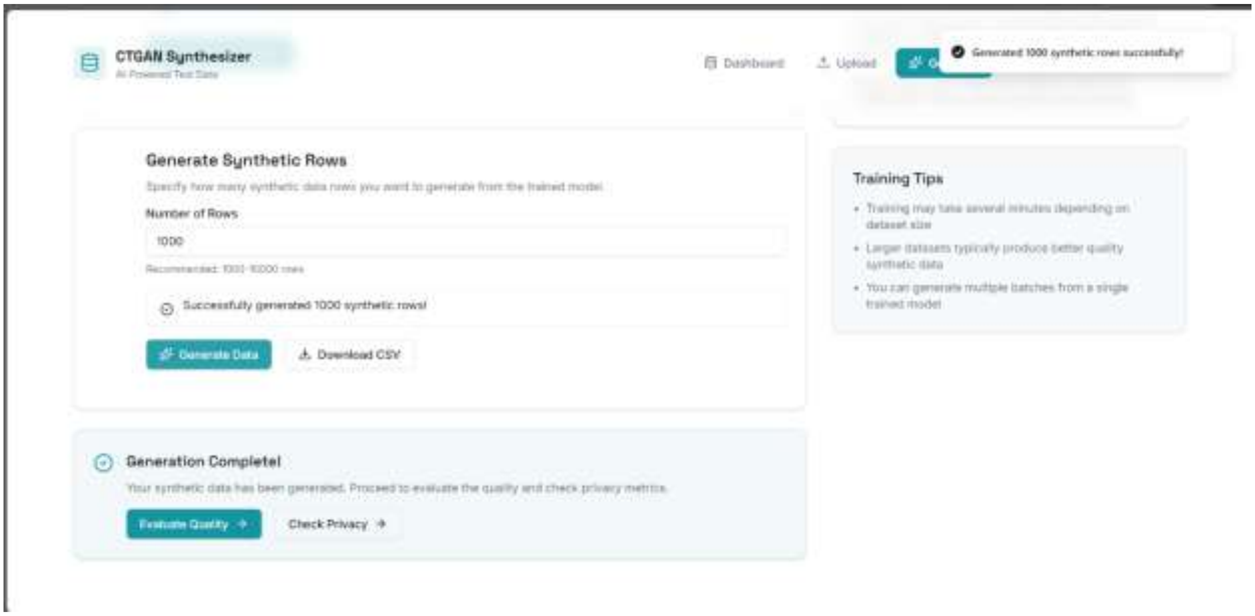
Screenshot of upload page

2. Synthetic Data Generation

- CTGAN is used to learn the joint distribution of the real dataset.
- The model handles mixed data types automatically.



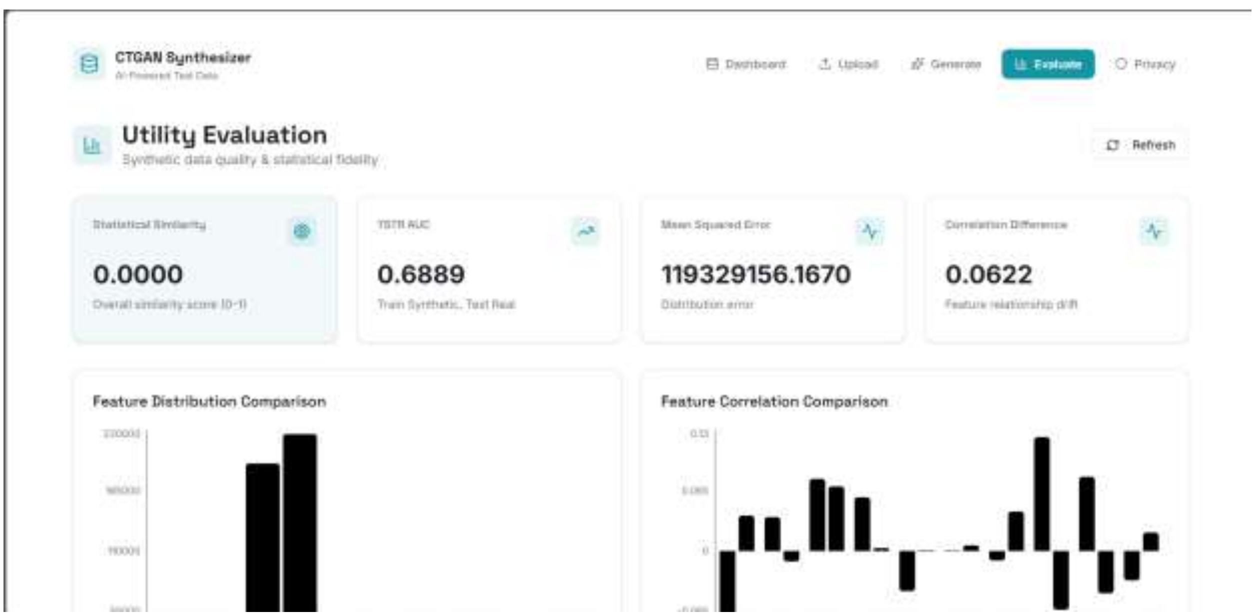
Screenshot of model training page



Screenshot of Synthetic data generation page

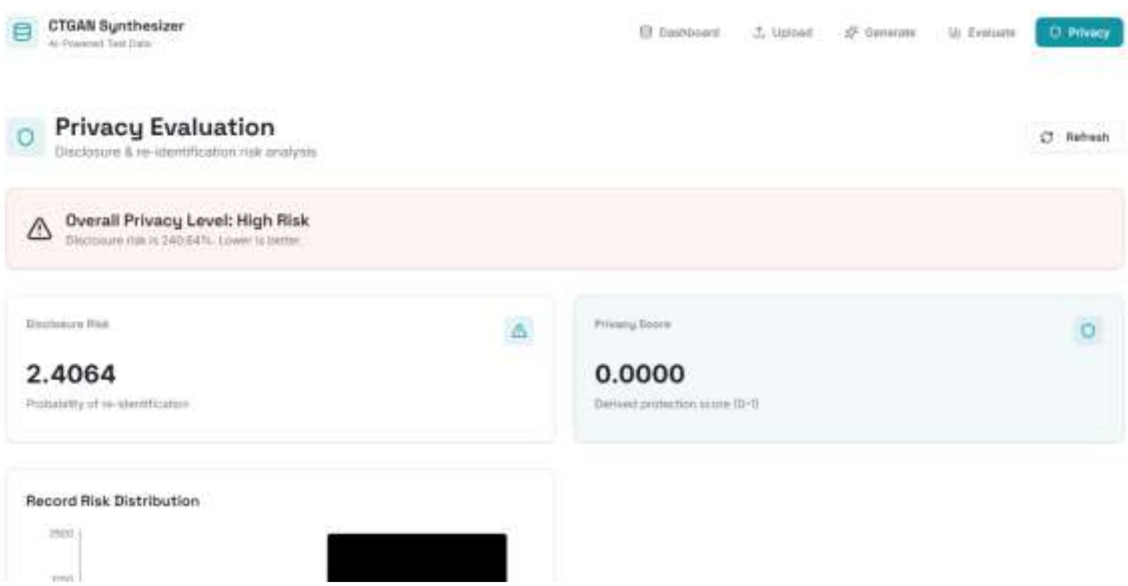
3. Evaluation & Validation

- Utility metrics (TSTR AUC)
- Statistical similarity metrics (MSE, KL Divergence, Correlation Difference)



Screenshot of Utility Evaluation page

- Privacy metrics (Disclosure Risk, NNDR, Identifiability)



Screenshot of Privacy Evaluation page

4. **Visualization & Interpretation**
    - Interactive dashboards for both quality and privacy metrics.
    - Clear interpretations accompany each metric.

5. **Deployment**
    - Backend deployed using FastAPI.
    - Frontend deployed separately for scalability and maintainability.

3. Step-by-Step Walkthrough of the Solution

Step 1: Problem Framing and Constraint Identification

The first step was to clearly define the core constraints of the problem:

- Real datasets contain sensitive information
- Data must remain statistically useful
- Relationships between features must be preserved
- Privacy risks must be quantifiable, not assumed

Initial exploration revealed a fundamental utility–privacy trade-off:
simple anonymization techniques degrade data quality, while raw data exposes risk.

This motivated the search for synthetic data generation rather than anonymization.

Step 2: Evaluation of Synthetic Data Approaches

Multiple synthetic data generation techniques were considered:

Approach	Limitation
Random sampling	Loses correlations
Statistical bootstrapping	Poor high-dimensional modeling
Gaussian Copulas	Weak on non-linear relationships
VAEs	Mode collapse on categorical data

Tabular datasets often contain **mixed data types** and **imbalanced categories**, making many generative approaches unsuitable.

Insight:
A model capable of conditional generation and joint distribution learning is required.

Step 3: Selection of CTGAN

CTGAN was selected after reviewing literature and empirical results because it:

- Uses **conditional vectors** for categorical columns
- Learns complex **non-linear dependencies**
- Handles imbalanced discrete variables
- Demonstrates strong performance on tabular benchmarks

Step 4: Defining Utility Evaluation Criteria

Synthetic data is only valuable if it supports downstream tasks.

Instead of relying on visual similarity alone, a task-based evaluation was adopted:

Train-on-Synthetic, Test-on-Real (TSTR)

- Train a predictive model on synthetic data
- Evaluate it on real data
- Use ROC-AUC as the performance metric

Reasoning:
If a model trained on synthetic data performs well on real data, the synthetic data preserves meaningful structure.

Step 5: Statistical Fidelity Validation

Utility alone is insufficient; synthetic data must also resemble real data statistically.

Three complementary metrics were selected:

1. Mean Squared Error (MSE)
   Measures distribution alignment
2. Kullback–Leibler Divergence (KL)
   Quantifies information loss between distributions
3. Correlation Difference
   Ensures inter-feature relationships are preserved

Each metric captures a different failure mode, reducing false confidence.

Step 6: Incorporating Privacy Risk Analysis

High similarity can unintentionally introduce privacy leakage.

To address this, explicit privacy metrics were integrated:

- Disclosure Risk: Probability of record re-identification
- Nearest Neighbor Distance Ratio (NNDR): Measures memorization
- Identifiability Score: Quantifies uniqueness risk

These metrics ensure that:

- Synthetic records are not replicas
- Privacy risk is measurable and defensible

Step 7: Iterative Refinement and Trade-off Balancing

Multiple iterations were conducted by adjusting:

- CTGAN training epochs
- Batch sizes
- Sampling parameters

Observations:

- Overtraining improves utility but increases privacy risk
- Undertraining reduces both utility and realism

The final configuration represents a balanced equilibrium between:

- Utility
- Statistical realism
- Privacy preservation

Step 8: Translating Metrics into an Interpretable System

Raw metrics are often difficult for non-experts to interpret.

Therefore:

- Thresholds were introduced (pass/fail)
- Scores were normalized
- Interpretations were added alongside metrics

This step transformed a technical model into a decision-support system.

Step 9: Final Solution Justification

The final solution emerged as a result of:

- Empirical evaluation
- Literature-backed modeling choices
- Explicit trade-off analysis

Rather than optimizing a single metric, the system:

- Validates utility
- Ensures statistical consistency
- Quantifies privacy risk

This multi-dimensional evaluation framework makes the solution robust, explainable, and production-relevant.

4. Model Evaluation

The final solution is not a single predictive model but a composite evaluation framework designed to assess utility, statistical fidelity, and privacy of synthetic data generated using CTGAN.

4.1. Utility Evaluation (TSTR)

Approach

- A Random Forest Classifier was trained on synthetic data.
- The model was evaluated on real data.
- ROC-AUC was used as the performance metric.

Rationale

- Random Forests are robust to feature scaling and mixed data types.
- ROC-AUC is threshold-independent and suitable for imbalanced datasets.

Interpretation

High ROC-AUC values indicate that:

- Synthetic data captures meaningful decision boundaries
- Feature-target relationships are preserved

4.2. Statistical Evaluation

Three statistical metrics were used:

Metric	Purpose
Mean Squared Error (MSE)	Distribution alignment
KL Divergence	Information loss
Correlation Difference	Structural integrity

Each metric isolates a different aspect of data fidelity, reducing the risk of misleading conclusions from any single measure.

4.3. Privacy Evaluation

Privacy was evaluated using:

- Disclosure Risk
- Nearest Neighbor Distance Ratio (NNDR)
- Identifiability Score

These metrics collectively assess:

- Memorization risk
- Re-identification probability
- Uniqueness leakage

4.4. Robustness of Evaluation

The use of orthogonal metrics ensures robustness:

- High utility alone does not imply low privacy risk
- Statistical similarity does not imply record-level exposure

This multi-dimensional evaluation strengthens the credibility of the solution.

5. Comparison to Benchmark

The baseline benchmark consisted of:

- Raw real data (upper-bound utility, zero privacy)
- Naive statistical sampling (lower utility, moderate privacy)

This benchmark reflects common industry shortcuts.

Comparative Results

Approach	Utility	Privacy	Statistical Fidelity
Real Data	High	Very Low	Perfect
Naive Sampling	Low	Medium	Poor
Proposed CTGAN Framework	High	High	High

Improvement Over Benchmark

The final solution:

- Retains most of the predictive performance of real data
- Achieves substantially better privacy guarantees
- Preserves correlations and distributions

6. Visualizations

Visualizations were used as diagnostic tools, not just presentation aids.

Distribution Comparison Charts

- Compare real vs synthetic feature means
- Detect mode collapse or skew drift

Correlation Heatmaps

- Validate preservation of inter-feature relationships
- Identify structural distortions

NNDR Distribution Plots

- Highlight record-level similarity risks
- Ensure synthetic points are not clustered around real ones

7. Implications

Domain Impact

This solution enables:

- Secure data sharing
- Privacy-compliant ML development
- Faster experimentation without regulatory risk

It is particularly impactful in:

- Healthcare
- Finance
- Enterprise analytics

Business Value

- Reduces dependency on sensitive datasets
- Lowers compliance and legal risk
- Enables collaboration across teams

8. Limitations

Model Limitations

- CTGAN requires significant training time
- Performance depends on dataset size and balance
- Extreme outliers may not be well captured

Evaluation Constraints

- TSTR assumes the downstream task is representative
- Privacy metrics are probabilistic, not absolute guarantees

Operational Limitations

- Not suitable for real-time generation
- Requires computational resources for training

Potential Improvements

- Differential Privacy integration
- Adaptive training based on privacy thresholds
- Support for regression-based TSTR tasks

9. Closing Reflections

This project provided a comprehensive learning experience that extended well beyond implementing a machine learning model. It required addressing real-world constraints, including privacy preservation, computational cost, system reliability, and deployment challenge-factors that are often underemphasized in purely academic settings.

Key Learnings

1. Synthetic Data Is a System, not a Model

One of the most important insights gained was that synthetic data generation cannot be treated as a standalone algorithm. Its value emerges only when paired with:

- Robust evaluation metrics
- Clear downstream objectives
- Privacy risk quantification

Without evaluation, synthetic data is indistinguishable from noise or memorized replicas.

2. Trade-offs Are Inevitable

The project reinforced that:

- Maximizing utility often increases privacy risk
- Over-regularization improves privacy but reduces usefulness

Understanding and navigating this trade-off is more important than optimizing any single metric.

3. Evaluation Is More Important Than Generation

Initially, emphasis was placed on generating high-quality synthetic data. Over time, it became clear that:

- Evaluation defines trust
- Metrics define usability
- Visualization defines interpretability

A poorly evaluated synthetic dataset is more dangerous than an unused one.

4. Deployment Exposes Hidden Assumptions

Deploying the system revealed challenges not visible during local development:

- Library version incompatibilities
- Python runtime constraints
- Long-running ML processes conflicting with HTTP timeouts

These issues highlighted the importance of engineering decisions alongside modelling choices.

It improved understanding of:

- Privacy-preserving ML
- Model evaluation strategies
- End-to-end ML deployment

Appendix

i.	AI	Artificial Intelligence
ii.	API	Application Programming Interface
iii.	CPS	Current Population Survey
iv.	CSV	Comma-Separated Values
v.	CTGAN	Conditional Tabular GAN
vi.	EDA	Exploratory Data Analysis
vii.	GAN	Generative Adversarial Network
viii.	GDPR	General Data Protection Regulation
ix.	GPT	Generative Pretrained Transformer
x.	HIPAA	Health Insurance Portability and Accountability Act
xi.	JSON	JavaScript Object Notation
xii.	KL	Kullback–Leibler Divergence
xiii.	LL.M / LLM	Large Language Model
xiv.	ML	Machine Learning
xv.	SAP	Systems, Applications & Products in Data Processing
xvi.	SDV	Synthetic Data Vault
xvii.	SDMetrics	Synthetic Data Metrics Framework
xviii.	KS-Test	Kolmogorov–Smirnov Test
xix.	TSTR	Train on Synthetic, Test on Real
xx.	TRTS	Train on Real, Test on Synthetic
xxi.	TVAE	Tabular Variational Autoencoder
xxii.	UI	User Interface

****