# Summer Internship Report: Exploration of Databricks Unity Catalog Features and Development of a Voice-Enabled Query Interface

## 1. Introduction

This report summarizes the work undertaken during my two-month summer internship. The internship focused on exploring Databricks Unity Catalog governance features and contributing to the development of a speech-to-query interface that enables natural language interaction with data systems. The work involved both theoretical exploration and practical implementation using open datasets.

## 2. Objectives

- Learn and experiment with Databricks Unity Catalog’s governance features: Data Discovery, Data Lineage, and Auditing.  
- Apply these features on open datasets to understand their practical applications.  
- Contribute to designing a speech-to-query system for non-technical users to access data.

## 3. Work Summary

### 3.1 Databricks Unity Catalog Feature Exploration

The Unity Catalog feature exploration focused on three major governance capabilities:

• Data Discovery: Enabled tagging, metadata management, natural language search, and integration with diverse systems.  
• Data Lineage: Provided end-to-end tracking of data transformations, impact analysis, and column-level traceability.  
• Auditing: Offered detailed access logs, automated compliance reporting, and anomaly detection for data security.

Hands-on implementation was performed using open healthcare datasets such as Synthea Synthetic Health Records, PharmGKB Clinical Annotations, and COVID-19 public data. These exercises demonstrated how governance features streamline data management and improve compliance.

### 3.2 Speech-to-Query Generation Project (DataScope)

The speech-to-query project, named 'DataScope', aimed to allow clinicians and analysts to query datasets without learning SQL or accessing Databricks directly.

Solution Overview:

1. Voice input captured via browser SpeechRecognition API.  
2. Conversion of speech to text, followed by natural language to SQL translation using an LLM model (Meta LLaMA-3).  
3. Execution of SQL queries via FastAPI backend connected to Databricks SQL endpoint.  
4. Role-based access controls and PHI/PII masking applied before returning results.

Due to company policy, the source code cannot be shared. This section presents only the architecture and high-level workflow.

## 4. Skills and Tools Learned

• React.js for frontend development.  
• FastAPI for backend API creation.  
• SQL for data management.  
• Unity Catalog metadata models.  
• LLM prompt engineering and fallback mechanisms.

## 5. Challenges Faced

• Limited direct access to sensitive Databricks datasets.  
• Ensuring accurate and reliable SQL generation from voice queries.  
• Maintaining compliance while designing the prototype.

## 6. Outcomes

• Demonstrated Unity Catalog governance features using open datasets.  
• Developed a functional speech-to-query interface for internal use.  
• Gained practical insights into enterprise data governance and AI integration.

## 7. Conclusion

The internship provided valuable hands-on experience in both modern data governance systems and AI-powered interfaces for data access. The combination of Databricks Unity Catalog exploration and speech-to-query project work allowed for the application of theoretical knowledge to practical, impactful solutions.