ADS 507 – Final Project – Team 2

Design Document

GitHub: [Pii-USD/ADS507 (github.com)](https://github.com/Pii-USD/ADS507)

Marvin Moran

Muris Saab

Ravita Kartawinata

# Source Datasets

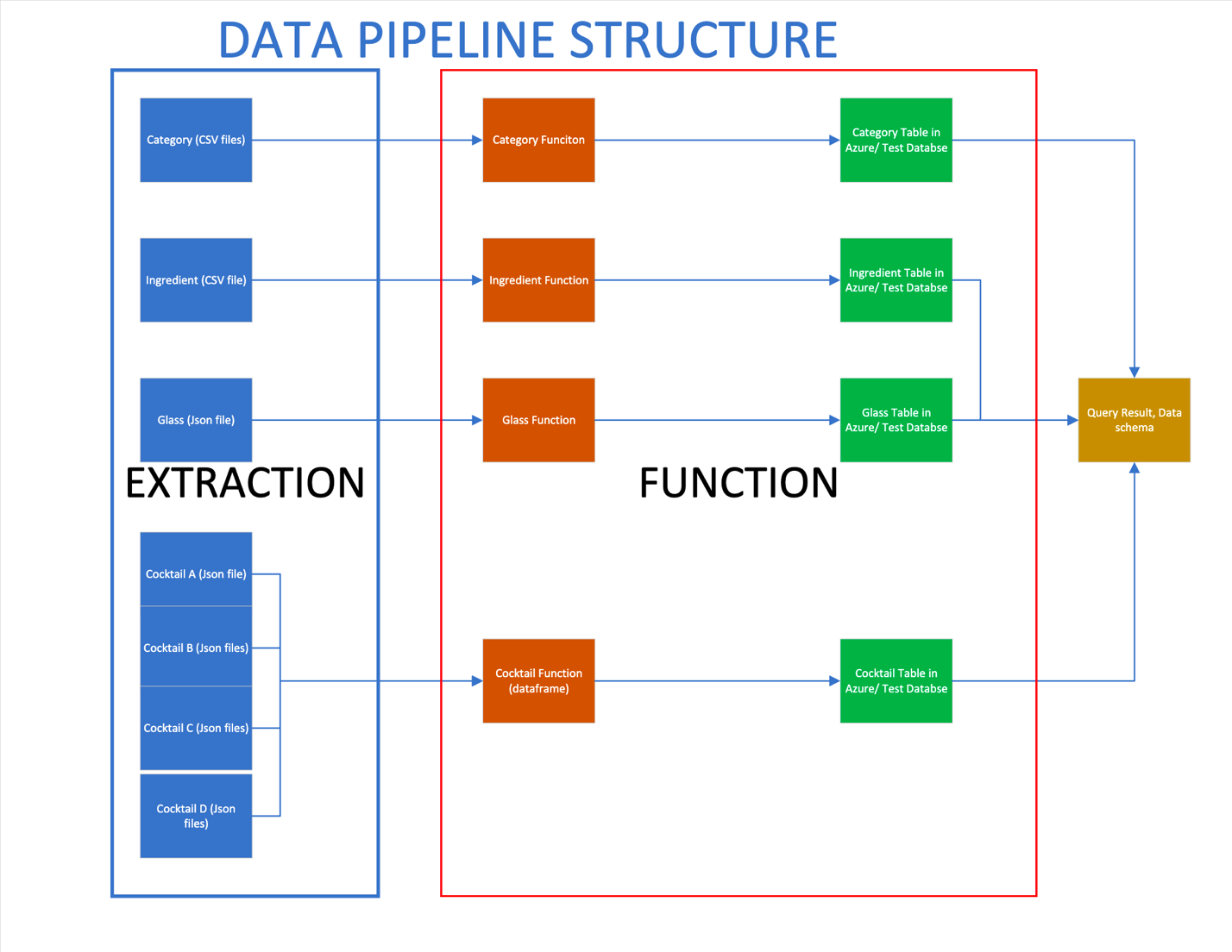
The datasets being used are originated from the following sources. They we selected because they allowed us to implement the concepts and techniques we learned during the course.

|  |  |
| --- | --- |
| Source | Datasets |
| Drinks API <https://www.thecocktaildb.com/>  An open, crowd-sourced database of drinks and cocktails from around the world. | Glass and  Drink tables |
| Comma Separated Values (CSV) files  These CSV files were generated to represent the concept of multiple sources. | Category and  Ingredients |

# Goals

The goal is to ingest the data for alcoholic and non-alcoholic beverages via an Extract, Transform, and Load (ETL) approach that will generate a final SQL dataset. The final dataset will then be used to support a searchable catalog for users.

# Pipeline diagram



# Steps to deploy pipeline

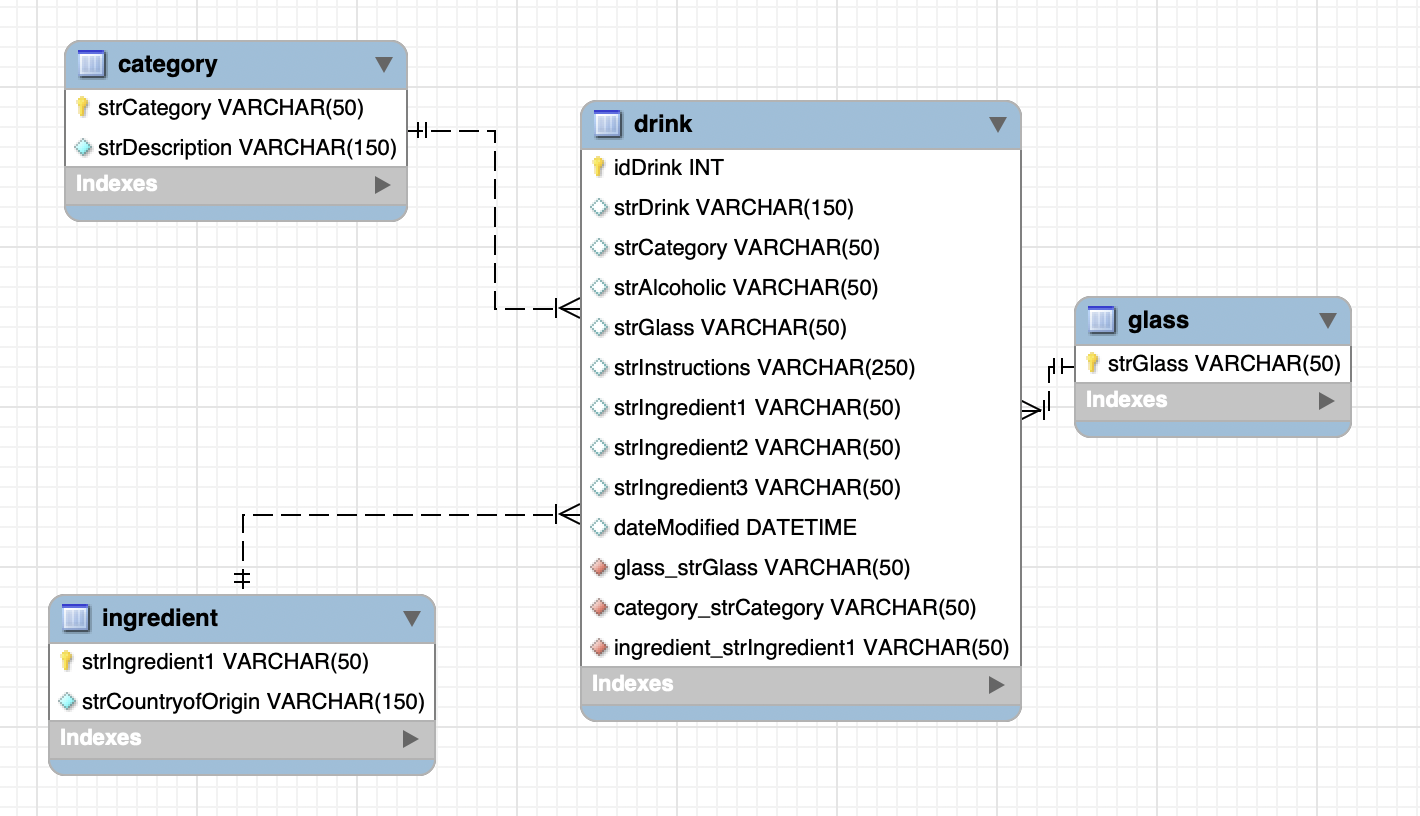
1. Ensure you have a local instance of MySqlWorkbench running
2. Download drinks.ipynb and the 2 CSV files to the same folder
3. Open the drinks.ipynb notebook file
4. Update the connection parameters in the first code block to your local instance of MySqlWorkbench
5. Run all code blocks in sequence
   1. First, the code will ensure that the necessary libraries and packages are downloaded to create a connection to the Azure data store. Also, the require parameters for creating a connection to the data store are set within the first code cell.
   2. The subsequent code cell then creates the connection engine utilizing the sqlalchemy package.
   3. Next, the following code cells connect to the Drinks API noted above and extract data in a JSON format.
   4. Similarly, the following code cells thereafter ingests the 2 target CSV files.

Following these code executions, the next code cells allow user to:

1. trigger the input process in which they submit a value (e.g., ingredient = 'vodka').
2. trigger the query execution and results process.

# Schema

Supporting data is present in this repository, both as CSV files and a JSON file. The 'category.csv' file and 'ingredients' file are the 2 target CSV files while the 'drinkDB\_schema.mwb' is the target JSON file.



# Output validation

The pipeline output is validated by a sample code running Step 4 in the Jupiter notebook:

In this example, if we select ‘beer’ glass, the output table can be queried for a simple EDA analysis of the category:

A graph of blue rectangular bars

Description automatically generated with medium confidence

# Considerations and known gaps

This pipeline was designed for educational purposes. We aimed to enable any student running a local instance of MySqlWorkbench to reproduce the same steps and learn about the process.

The dataset is both small and primarily static, which means that the system resources required to process it are minimum. The frequency of updates to the upstream data sources is very low so no automated ingestion mechanisms were considered. No growth is expected for this dataset while it is used as a proof of concept.

From a security perspective, the use of a local instance of MySqlWorkbench also assume the use for educational purposes, otherwise the use of hard coded credentials in code is not be recommended.