# Data Warehousing (CSE 609)

Spring Semester 2024

**Health Care Industry Analysis**

**Project Report**

**Group Members**

|  |  |
| --- | --- |
| **Name** | ERP ID |
|  |  |
| **Talha Sarwar** | **29458** |
|  |  |
| **Murtaza Hussain** | **29449** |
|  |  |

***Submitted to: Dr. Tariq Mahmood***

***Date of Submission: 2 June 2024***

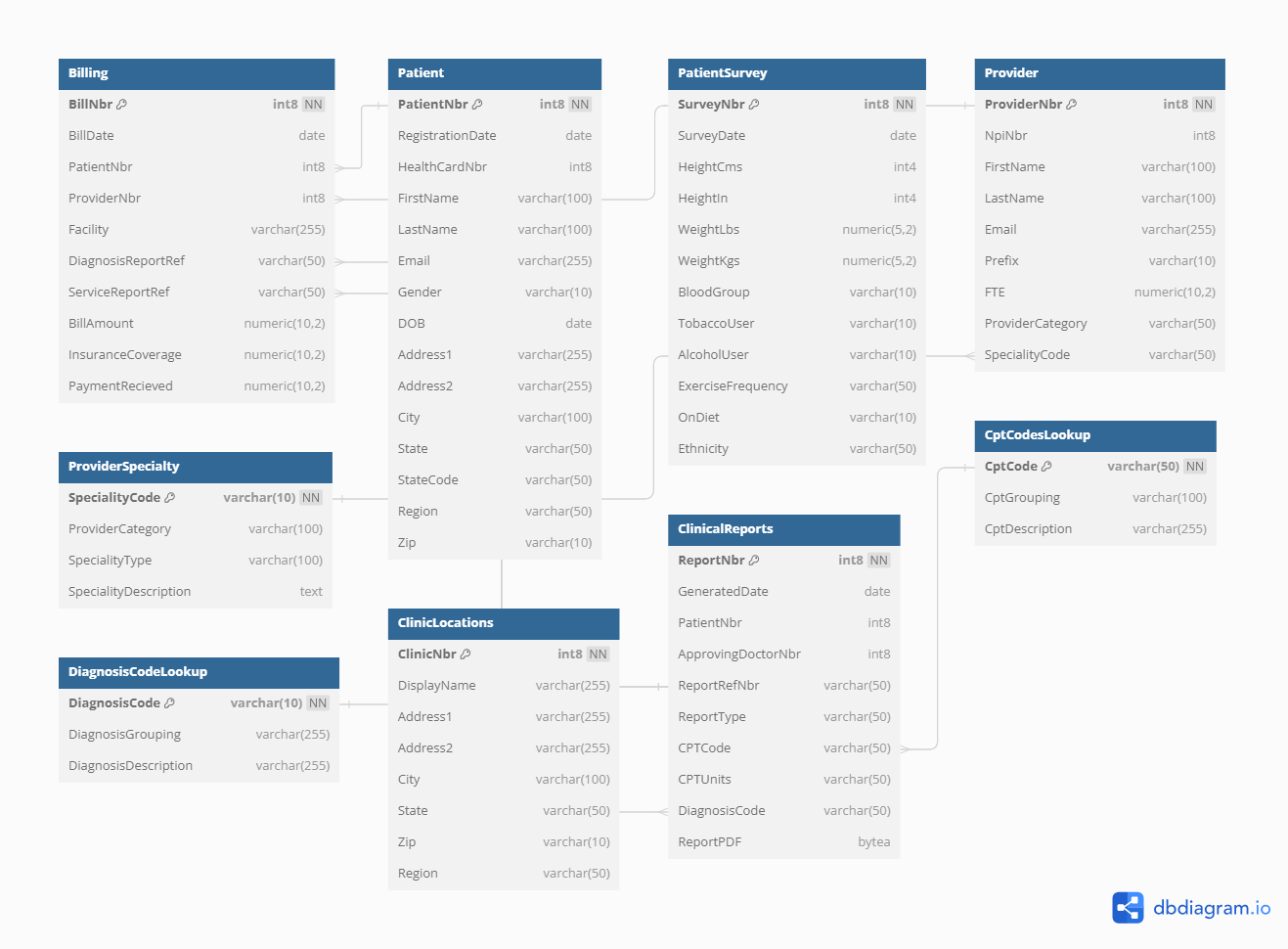
**Tools Used:**

|  |  |
| --- | --- |
| **Tools** | **Puprpose** |
| Python | For ETL pipeline Creation |
| Pandas | for processing and transformation of data |
| Postgres | To store and retrieve OLTP and OLAP Tables Data |
| CSV | To fetch OLTP Data |
| Airflow | For ETL Pipeline Orchestrationa and Scheduling and defining dependencies |
| PowerBI | To create the dashboard |
| Linux | To setup postgres and airflow |

1. **The Database Spaghetti:**

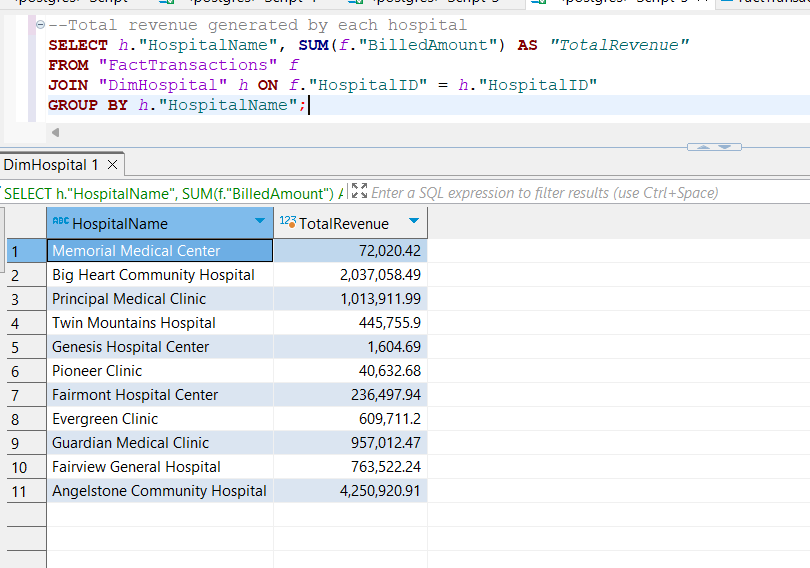
* We Took help from HealthCare Industry Experts to understand the table and attributes structure in healthcare industry, we created the tables and attributes according to it and tried our best to produce dummy data  
  We used some python scripts to create dummy data for some columns  
  For other columns we referred to Kaggle website for the data  
  Scripts are available in below folder  
  **Folder Name:** Data Generator Scripts
* OLTP tables csv available in “EXC” zipped folder
* OLTP Tables list and little description of each Table

|  |  |
| --- | --- |
| **Table Name** | **Description** |
| **Billing** | Stores transaction details related to billing for healthcare services, including bill numbers, patient numbers, provider numbers, bill dates, amounts, and payments. |
| **ClinicLocations** | Contains details of healthcare facilities or clinics, including clinic numbers, display names, addresses, cities, states, zip codes, and regions. |
| **ClinicalReports** | Holds clinical report data, such as diagnosis and service reports, including diagnosis codes, CPT codes, report references, and generated dates. |
| **CptCodesLookup** | Provides a reference for CPT (Current Procedural Terminology) codes, including CPT codes, their descriptions, and groupings. |
| **DiagnosisCodeLookup** | Contains a reference for diagnosis codes, including diagnosis codes, descriptions, and groupings. |
| **Patient** | Stores patient information and demographic details, including patient numbers, registration dates, health card numbers, names, emails, gender, age, date of birth, addresses, and contact details. |
| **PatientSurvey** | Holds additional health-related information collected through patient surveys, including details on height, weight, blood group, tobacco and alcohol use, exercise frequency, diet, and ethnicity. |
| **Provider** | Contains information about healthcare providers, including provider numbers, NPI numbers, prefixes, names, emails, full-time equivalency (FTE), and provider categories. |
| **ProviderSpecialty** | Provides details about the specialties of healthcare providers, including specialty codes and types. |

* Spaghetti Database ERD  
  > We created it using dbdiagram.io, code to create is available in below word file   
  File Name: ERD/ERD of Spaghetti and Star Schema.docx  
  
* Python script to load data from CSV files data to postgres DB  
  **File name:** PYT/Excel To OLTP DAG.py

1. **DWH Design Excel Template (EXC):**- Template File name: EXC\_Template\_HealthCare\_Domain.xlsx
2. **Pipeline Python Script (Conversion Functions) (PYT):**

* Script File Name: PYT/OLTP To OLAP DAG.py
* We created the functions for each table separately.
* We fetched data from postgres OLTP schema, staged it in pandas dataframe and then write the data back in OLAP Schema

1. **Pipeline Python Script (Dimensional Query Functions) (PYT):**- As we are using linux OS for the execution of scripts, so we used Query editor to connect with postgres and then there we executed our scripts on it, Screenshot of outputs are attached against every query  
   ****  
   **A screenshot of a computer

   Description automatically generated**  
   **A screenshot of a computer

   Description automatically generated**  
   **A screenshot of a computer

   Description automatically generated**  
   **A screenshot of a computer

   Description automatically generated**
2. **Pipeline Python Script (Generation of Fact Table Snapshot) (PYT)**

* **Script File Name:** PYT/Star Schema To Fact Table Snapshot DAG.py

1. **Dashboarding (PowerBI + PYT):**- We used powerBI Desktop  
   - As we are using linux to run our scripts, so we cannot integrate it with Python as we cannot have a visual like that  
   - We connected powerBI with our postgres DB which we have installed on linux server  
   A screenshot of a computer

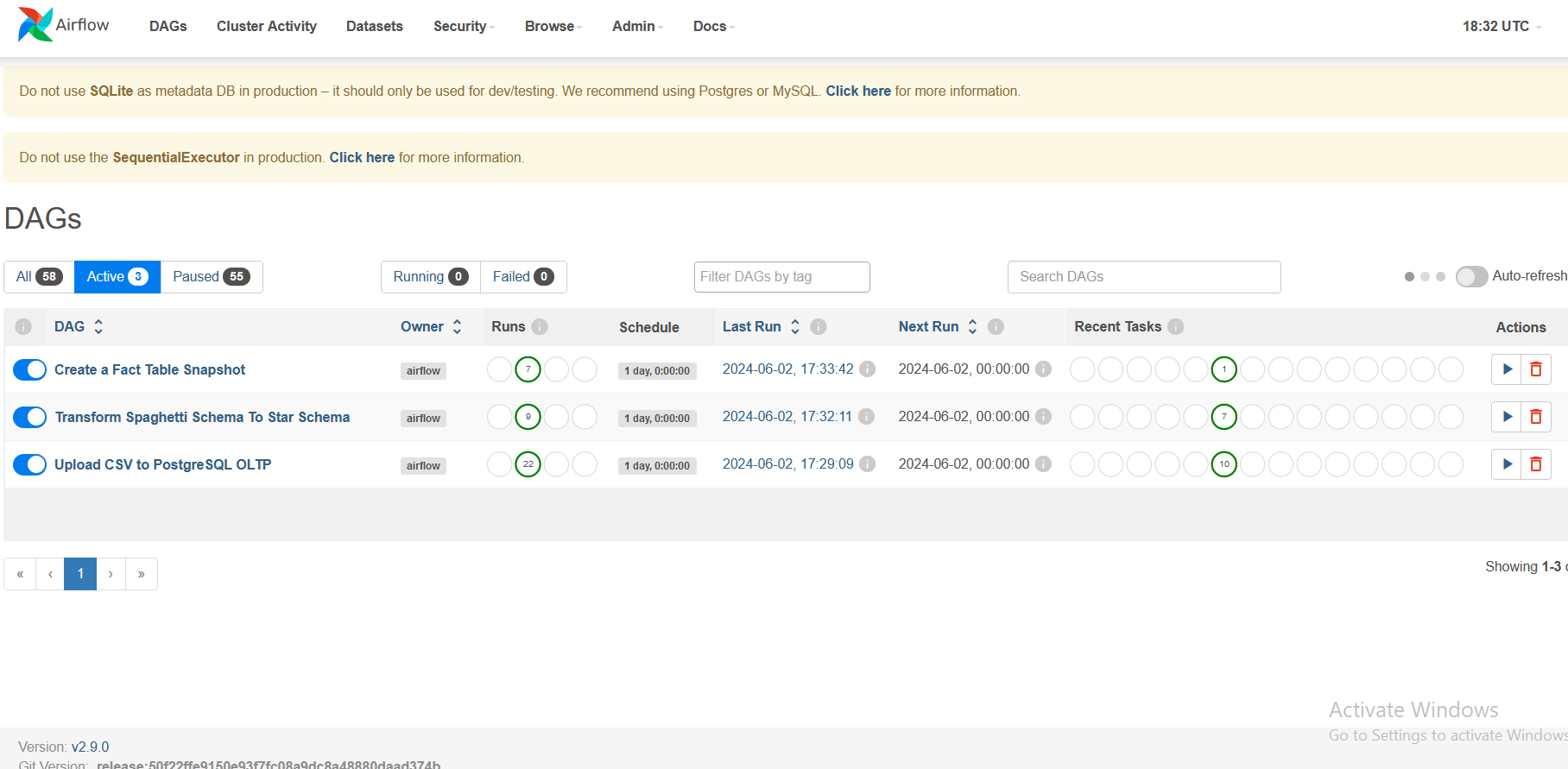
   Description automatically generated  
   - We have then posted it online so that others can access it  
   **Link:** https://app.powerbi.com/view?r=eyJrIjoiNTljZDA5OTMtZTBjYy00ODM2LTliM2YtMzEwZWRmMjY0MmQ3IiwidCI6ImZlZTNiOTE2LTAxYzEtNDk4Ny1hNjQ2LWUxOTM0MzJiOWVhYSIsImMiOjl9  
   - We have created 3 pages in our Dashboard  
   **1st page**  
   A screenshot of a graph

   Description automatically generated

**2nd Page  
A screenshot of a medical report

Description automatically generated  
  
3rd Page  
A screenshot of a patient outcome analysis

Description automatically generated**

1. **Automation and Monitoring (PYT)**- We have used airflow to create a ETL pipline and to orchestrate ETL jobs through it- Below is a snap of 3 of our workflows, inside which there are different tasks and each tasks represent loading data into a specific table ****  
   - using variables option in airflow we are controlling the run date for our pipeline and we can also set a frequency through it  
   **A screenshot of a computer

   Description automatically generated**
2. **A design diagram of your pipeline (WRD) – with any comments or observations**A diagram of a software application

   Description automatically generated with medium confidence
3. **Submission of a video demo of your complete project.** The video should not be more than 5 minutes.

**File Name:** Video/Project\_Demo\_Video