**Technical Documentation – Hospital Analytics**

**Personal project by**: Aymane RAMI  
**Year**: 2025

**Project Objective**

The goal of the *Hospital Analytics* project is to simulate a complete data pipeline for a fictional hospital. This includes generating healthcare data, ingesting it into a cloud data warehouse (**Snowflake**), transforming it using **dbt**, and finally visualizing the results in **Tableau**.

This project covers the full data lifecycle:

* Simulated data generation (patients, doctors, hospitalizations...)
* Data ingestion into a warehouse
* Data modeling and transformation using dbt
* Export and visualization of the results
* Preparing for future pipeline automation

**Technologies Used**

**Languages and Tools**

* **Python**  
  Used to generate fake data, load CSV files into Snowflake, and interact with the database via APIs.
* **Snowflake**  
  Cloud-based data warehouse to store raw and transformed data. Configured using SnowSQL.
* **dbt (Data Build Tool)**  
  Open-source tool for SQL-based data transformation, modeling, and documentation.
* **Tableau Desktop (free version)**  
  Used to create initial dashboards from the SQL views generated by dbt.
* **SnowSQL**  
  Command-line interface for interacting with Snowflake: table creation, file loading, export, etc.
* **Pandas, Faker, NumPy (Python)**  
  For data manipulation, random data generation (Faker), and structured file creation.
* **Git & GitHub**  
  Version control and future CI integration.

**Repository Structure**

hospital\_analytics/

├── data/

│ └── csv/

│ ├── patients.csv

│ ├── doctors.csv

│ ├── hospitalizations.csv

│ ├── admissions\_by\_month.csv

├── data\_generator/

├── generate\_fake\_data.py

├── dbt/

└── hospital\_analytics/

│ └── logs/

│ └── models/

│ ├── patients\_demographics.sql

│ ├── patients\_admissions\_by\_month.sql

│ ├── patients\_hospitalizations.sql

│ └── dbt\_project.yml

└── venv/

└── README.md

**Project Steps**

**1. Simulated Data Generation (Python)**

A Python script (generate\_data.py) produces realistic hospital data:

* **Patients** (name, age, gender, admission date…)
* **Doctors** (name, specialty…)
* **Hospitalizations** (length of stay, diagnosis, etc.)

The data is generated using the **Faker** library, saved to CSV files, and structured to maintain referential integrity (IDs, foreign keys...).

**2. Loading Data into Snowflake**

* Connection to Snowflake via the **snowflake-connector-python**
* Table creation inside the HOSPITAL\_ANALYTICS\_SCHEMA schema of the HOSPITAL\_ANALYTICS\_DB database
* Internal **STAGE** creation for CSV loading
* Upload via the PUT command (SnowSQL)
* Data insertion via COPY INTO into the final tables

**3. Modeling with dbt**

The hospital\_dbt project was initialized to handle SQL transformations in Snowflake.

**Configuration**

* profiles.yml was configured with:
  + Snowflake account name
  + Username and password
  + Database, schema, and warehouse (COMPUTE\_WH)

**Models Created**  
SQL models in hospital\_dbt/models/:

* patients\_demographics.sql: count of patients by gender
* patients\_admissions\_by\_month.sql: number of admissions per month
* patients\_hospitalizations.sql: average stay duration

These models simplify future dashboarding.

**Execution**

* Compile transformations with dbt run
* Documentation generated using dbt docs generate (not hosted yet)

**4. Exporting Results from Snowflake**

The views created by dbt were exported using COPY INTO @~/export/ in .csv.gz format.

**5. Data Visualization with Tableau**

Extracted CSV files were imported into **Tableau Desktop (Free)** to build a basic dashboard showing:

* Monthly admissions (bar chart)
* Gender distribution (histogram)
* Average length of hospitalization

This simple dashboard can be extended for deeper analysis.

**Key Technical Notes**

* **SnowSQL Config**: config file was updated with proper credentials (account, user, warehouse…)
* **SnowSQL Installation**: installed via MSI to fix detection issues in terminal environments

**Next Steps**

**1. Full Pipeline Automation**

* Auto-deploy dbt models via **GitHub Actions**
* Daily export of analytics results
* Periodic refresh of simulated data

**2. Enhanced Tableau Dashboard**

* Live Tableau connection to Snowflake
* More dbt models for richer insights
* KPIs: readmission rate, average cost per patient, etc.

**3. Data Testing with dbt**

* Add quality tests (null values, referential integrity)
* Serve interactive documentation with dbt docs serve

**4. Automated Reporting**

* Generate PDF or HTML reports from dbt results
* Send via email using Python or GitHub Actions

**Conclusion**

The **Hospital Analytics** project implements a complete data analytics pipeline from simulated data to visualization.  
It demonstrates strong proficiency with modern data tools (**Snowflake**, **dbt**, **Tableau**) and scripting skills in Python.

This project offers a solid base for real-world use cases in healthcare analytics, business intelligence, or data engineering, and can be significantly extended for more complex needs.