Course Project
CS 236: Database Management Systems (Fall 2025)

TA: Ahmed Abdelmaguid
aabde039@ucr.edu

Contents

Overview & Goal

This course project is a quarter-long team project. You will work in teams of two to build a small analytics pipeline over two hotel reservation datasets (2015–2016 and 2017–2018).

The goal is to clean and unify the datasets using **PySpark**, compute key analytics, persist the cleaned and merged tables into a Dockerized **PostgreSQL** database, and implement a simple web UI to filter and query the data.

Outcomes

By the end of the project, you will have:

- Cleaned, consistent datasets unified into a single dataset.
- Key analytics computed using Spark.
- A PostgreSQL schema populated via Spark.
- A simple web interface to retrieve rows and apply filters over the unified dataset.

1 Phase 1: Data Preparation & EDA in Spark

1.1 Environment Setup

Install Docker

Docker will be mainly used in Phase 2, but it should be installed in this phase as well to ensure it is available and running.

If you already have Docker installed, verify it with:

```
docker --version
```

If not installed, follow the installation guide for your operating system: https://docs.docker.com/get-started/

Install and Run Spark

Install Spark on your local machine (not in Docker).

To install PySpark, run:

```
pip install pyspark==3.5.1
```

PySpark can be used either to run a Python script or through an interactive shell. It is recommended to use it for running scripts rather than editing interactively.

```
pyspark <python_file.py>
```

1.2 Exploratory Data Analysis (EDA)

Using PySpark, load the datasets, perform exploratory data analysis (EDA), and document your findings. The specific choice of EDA functions is up to you, but you need to justify the benefits and the insights gained from each function in the report.

- **Load & scan**: Load both CSVs using Spark.
- **EDA**: Explore the schema and perform quick checks: row counts, distinct values, nulls, distributions, etc ....

1.3 Dataset Processing

Apply the necessary transformations and fixes to merge both datasets into a single unified file.

- **Adjust the schema**: Map overlapping columns, resolve name/type mismatches, and create any required features for merging, **only if needed**.
- **Persist cleaned datasets**: Save each cleaned dataset for future use.
- **Merge**: Align columns to a consistent schema and produce a unified dataset.

The resolution of mismatches and inconsistencies, as well as the creation of new features, should be based on your own decisions and must be documented.

2 Phase 2: Spark Analysis and Database Population

2.1 Spark Analysis

Using PySpark, perform the following analyses on the unified dataset:

- **Cancellation rates**: Calculate cancellation rates for each month.
- **Averages**: Compute average price and average number of nights for each month.
- **Monthly bookings**: Count monthly bookings by market segment. In categories, the term `TA` means `Travel Agents` and `TO` mean `Tour Operators`
- **Seasonality**: Identify the most popular month of the year for bookings **based on revenue**.

2.2 Database Population

Start PostgreSQL with Docker

Run the following command to start a PostgreSQL container:

```
docker run -d \
  --name <container_name> \
  -e POSTGRES_USER=<username> \
  -e POSTGRES_PASSWORD=<password> \
  -e POSTGRES_DB=<db_name> \
  -p <host_port>:<container_port> \
  postgres:16
```

Design and Populate the Database

- **Schema design**: Create a schema that matches each of the three datasets (two cleaned and one unified).
- **Population**: Load the datasets into PostgreSQL using PySpark.

When executing commands to link Spark with PostgreSQL, ensure you include the `JDBC` driver:

```
pyspark --packages org.postgresql:postgresql:42.7.3 <python_file.py>
```

3 Phase 3: WebUI for Searching and Filtering

Build a lightweight WebUI using the technology stack of your choice. The application should connect to your PostgreSQL database and query the relevant tables.

The WebUI should provide the following functionality:

- **Data retrieval**: Display rows from the available datasets (Dataset 1, Dataset 2, or the Unified dataset).
- **Filtering**: Allow filtering based on column attributes (e.g., `avg_price_per_room`, `booking_status`).

4 Deliverables

For each phase, you will submit a zipped file with the following

Phase 1 Deliverables

For Phase 1, you are required to submit:

- A report documenting all the steps you took, including:
  - the installation process,
  - the exploratory data analysis (EDA) with plots (if applicable),
  - explanations of the reasons for choosing specific EDA functions,
  - insights gained from the analysis, and
  - the decisions made to clean, process, and merge the data.
- Well-documented source code (with comments).
- Three datasets:
  - two cleaned versions of the original datasets, and
  - one newly merged dataset.
- A 5-minute video presentation explaining and demonstrating your work.

Phase 2 Deliverables

For Phase 2, you are required to submit:

- A report documenting all the steps you took, including:
  - the decisions made to calculate each of the required analyses,
  - the database created and the schema you designed to integrate the three datasets, and
  - code explanations for populating the PostgreSQL database with the datasets.
- Well-documented source code (with comments).
- SQL files generated from your PostgreSQL database.
- A 5-minute video presentation explaining and demonstrating your work.

Phase 3 Deliverables

For Phase 3, you are required to submit:

- A report documenting all the steps you took, including:
  - demonstrations of the webpage search and filtering features, and
  - code explanations for the implementation of each feature, including how they connect to the PostgreSQL database.
- Well-documented source code (with comments).

In addition, you will need to prepare a short live demo to present your work.

5 Project Timeline

Start Date	Due Date	Task
Oct 1	Oct 14	Project – Phase 1
Oct 15	Nov 4	Project – Phase 2
Nov 4	Nov 25	Project – Phase 3

Demos and Presentations will be on Nov 26 to Nov 30 (none on Thanksgiving)

6 Teams

For this project, you will work in teams of two. The form for team registration will be released

7 AI Use Policy

You may use generative AI tools (e.g., ChatGPT) to support learning, especially for researching new tools, clarifying concepts, and viewing small examples. Your work must be clearly written and coded by you. Be prepared to show drafts/prompts, explain your work, and identify where AI was used and why.

Not allowed: Submitting AI output as your own. Direct copying is plagiarism and violates UCR Academic Integrity. Misuse will be treated as academic misconduct under UCR policy.