Department of Electrical and Computer Engineering
University of Puerto Rico
Mayagüez Campus

CIIC 4060/ICOM 5016 – Introduction to Database Systems
Spring 2024

Term Project – Hotel Analytics Systems
Phase III – Frontend
Due Date: April 25, 2024

Objectives

1. Understand the design, implementation and use of an application backed by a database system.
2. Understand the use of the E-R model for database application design.
3. Gain experience by implementing applications using layers of increasing complexity and complex data structures.
4. Gain further experience with Web programming concepts including REST.

Overview

You will design, implement, and test the backend of an application used to manage an inventory system. The data in the application is managed by a relational database system and exposed to client applications through a REST API. You will build the database application and REST API using **Flask**, which form the backend of the system. Your database engine must be **Postgres**, and you must implement the code in Python. The backend site will provide the user with the features specified in this document. In addition, your solution will provide a Web-based dashboard using **Voila** indicating relevant statistics that are also specified below.

Your solution MUST follow the Model-View-Controller Design Pattern. In this scheme, your solution will be organized as follows:

1) View – application pages will handle all interaction with the users and will show results from operations performed on the database. This is the client code for the application. The client **MUST NOT** interact directly with the database. They must talk through the REST API

2) Controller – **Python** objects will act as controllers. Each object will get a request, create a business service object to handle the request, collect the results from the methods in this business service object and forward the results to the client using JavaScript Object Notation (JSON).

3) Model – a set of business service objects that implement all tasks and access to the database system. **You cannot use ORM APIs for this layer**. **If your team uses ORM you will get an automatic 0 in the project.**

Details:

With Voila you will develop a frontend for the application. The necessary views for the frontend are:

a. Login page
    a. Must be able to login or create an employee account.
b. Local Statistics
    a. If user is employee, it can only see their own hotel; if user is supervisor must be able to choose any hotel of the same chain; if user is administrator must be able to choose any hotel in the database.
c. Global statistics
    a. Only accessible by administrator.
d. Create
    a. Administrator: must be able to create new record in the database, new records for any entity.
    b. Supervisors: must be able to create roomunavailable's records without reserve.
    c. Employees: must be able to create reservation's records.
e. Update & Delete
    a. Only administrators are able to update or delete records in the database.

All statistics needs a graph. The team can receive bonus points if the frontend is well designed. Buttons, dropdown bars, and other elements can be used to implement the functionalities.

You are required to use GitHub to manage and submit all phases' documents and code. You will be given access to a GitHub classroom link for this purpose.

Deliverables for Phase III

You will use the repo provided by classroom to submit the following:

1) Hosted database credentials (Should be created in **Heroku** and should have the tables)
2) Hosted REST API address (Use **Heroku**)
3) Code with the REST API and frontend (Use your respective repositories from Classroom)
4) Postman Collection with all the endpoints. (An endpoint for **each** of the routes in the app)
5) Functional frontend running on **Heroku dyno**.

PROJECT PHASE III DUE DATE: **11:59 PM – April 25, 2024.**

Oral Exams for this phase will be held, **tentatively**, **on May 8th-9th, 2024**

- You should bring your equipment to the Oral Exam