Computer Science and Engineering
University of Puerto Rico
Mayagüez Campus

CIIC 4060/ICOM 5016 – Introduction to Database Systems
Fall 2021

Term Project – Backend System for Google Calendar-Style Service
Phase 2 – REST API
Due Date: November 3, 2021, 11:59 PM

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 and HTTP.

Overview

You will design, implement, and test the backend of an application used to manage a booking system (will use Google Calendar as reference). 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 forms the backend of the system. Your database engine must be **relational**, 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 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 – applications, JavaScript pages, and HTML 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**.

As IDE, I recommend the use of JetBrains tools that fit your technology needs. In this case, PyCharm and DataGrip. (There is a way to connect the database directly to PyCharm without using DataGrip if you have the student version. For details visit Moodle.) Postman may aid your development process regarding testing.

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

Operations to be supported

Your site will support the following operations:

1. Register a new user
2. Find an available room (lab, classroom, study space, etc.) at a time frame
3. Find who appointed a room at a certain time
4. Give an all-day schedule for a room
5. Give an all-day schedule for a user
6. Create a meeting with 2+ people in a room
7. Limit the access to rooms appointment and information according to person's authorization (Professor, Student, Department Staff)
8. Find a time that is free for everyone in the meeting.
9. Allow user to mark time-space as "Unavailable"/ "Available" (By default it is all marked as available)
10. Only Department Staff can mark a time-space as "Unavailable"/ "Available" for any type of room (By default it is all marked as available)
11. CRUD operations for:
    a. Meeting
    b. Invitee
    c. User
    d. Room
12. User Statistic
    a. Most used Room
    b. User logged in user has been most booked with
13. Global Statistic
    a. Find busiest hours (Find top 5)
    b. Find most booked users (Find top 10)
    c. Find most booked rooms (Find top 10)

**CRUD: (Create, Read, Update, Delete)*

*Other functions or changes can occur in future phases.

For routes development use the following format:

https://<HOST>/<GROUP-NAME>/<ENTITY_RELATED_TO_OPERATION>/<NEEDED_INFO>

Deliverables for Phase II

In phase II you will use the repo provided by GitHub Classroom to submit the following:

1) Hosted database credentials (Should be created in Heroku and should have the tables)
    a. **This should be ready for October 21, 2021.**
2) Hosted REST API address (Use Heroku, DigitalOcean or other free alternatives)
3) Code with the REST API (Use your respective repositories from Classroom)
4) Postman Collection with all the endpoints. (An endpoint for each feature in the app)
5) ER and Table Diagram in PDF format. (If updated summit the new diagrams)

PROJECT PHASE I DUE DATE: **11:59 PM – November 3, 2021**.

Oral Exams for this phase will be held on **November 4th - 5th, 2021**

- You should bring your equipment to the Oral Exam.
- Bring a printed copy of your ER and Table Diagram.