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 3 – Frontend
Due Date: November 24, 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 and a basic frontend for all the operations.

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, WebStorm and DataGrip. 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 available rooms (lab, classroom, study space, etc.) at a time frame given user role.
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. Find a time that is free for every user given.
7. Create a meeting with 2+ people in a room
8. Limit the access to rooms appointment, information and schedule according to person's authorization. (Just for highest role)
9. Allow user to mark time-space as "Unavailable"/ "Available" in his schedule(should appear in user schedule and by default it is all marked as available)
10. Only highest can mark a time-space as "Unavailable"/ "Available" for any type of room (should appear in room schedule and By default it is all marked as available)
11. CRUD operations in the frontend:

Operation	Meeting	Invitee	User	Room
Create	Create a meeting	Just when the meeting is created	Sign Up	Just for the highest role
Read	As needed	As needed	As needed	As needed
Update	Can change meeting name or description.	--	Can change all but not his role	Just for highest role
Delete	User can delete a meeting if he is host	You can delete an invitee from a meeting	If it has no meeting or is not invitee	Just for highest role

12. User Statistic
    a. Most used Room by user given
    b. User most booked with user given
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)*

Views for application:

Your site will have the following views:

- Login/Register
- User View:
    - This view have functions divided by tabs. The number of tabs depends on the role and the functions that each role is allowed.
        - Creating meeting/unavailability
        - Schedule
        - Modify User information
        - User statistics
        - Other that may be needed to comply with the spect.
    - Buttons and other elements can be used to implements the functionalities.
    - Include Logout
- Dashboard for statistics

For routes development use the following format:
https://<HOST>/<GROUP-NAME>/<ENTITY_RELATED_TO_OPERATION>/<NEEDED_INFO>

Use this repository as a starter for your proyect. https://github.com/CIIC4060-ICOM5016-FALL-2021/db-booking-frontend . It has the instructions on how to run it. The libraries used in this project are Semantic, BigCalendar, and Recharts. You are required to used these libraries for the development of your project. You also have creative freedom as to how the views look, but it should maintain all the functionalities and can't go agaisn what is already required on this document.

The frontend should be included in your team's repository. (Instead of forking, download this project using zip and create a folder for it inside your already existing project. Your project structure should look something like this:

Project-
|
----frontend
|
----backend
|
----Documents
|
----README.md

Deliverables for Phase III

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)
2) Hosted REST API address (Use Heroku, DigitalOcean or other free alternatives)
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 feature in the app)
5) ER and Table Diagram in PDF format. (If updated summit the new diagrams)
6) Working frontend

PROJECT PHASE III DUE DATE: **11:59 PM – November 24, 2021**.

Oral Exams for this phase will be held on **December 7th - 8th, 2021**

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