

Universidad Interamericana de Puerto Rico Recinto de Arecibo

Departamento de Ciencias y Tecnología

Programa de Ciencias de Computadoras

Sistema de Gestión de Citas Médicas

COMP2052 – Web Development: Server Side & Microservices

Mikael Hashem K. Tiba (R00627615)

Janiel O. Rodríguez Velázquez (R00622992)

Descripción general del proyecto:

Este proyecto consiste en un sistema web de gestión de citas médicas desarrollado con Flask y MySQL. Permite a los usuarios registrarse e iniciar sesión, asignar roles como Administrador, Médico o Paciente, y realizar operaciones como agendar, cancelar y marcar citas como realizadas. Además, se implementaron pruebas de los endpoints con archivos .rest para validar el funcionamiento del CRUD completo.

Funcionalidades:

- Registro y autenticación de usuarios con Flask-Login.
- Asignación de roles: Administrador, Médico y Paciente.
- Agendamiento de citas por parte de pacientes, seleccionando fecha y médico.
- Cancelación de citas por pacientes, médicos o administradores.
- Marcado de citas como realizadas por médicos o administradores.
- Panel de control (dashboard) personalizado según el rol del usuario.
- Listado de usuarios (solo accesible para administradores).
- Cambio de contraseña seguro para cualquier usuario.
- Pruebas CRUD con endpoints REST (create, read, update, delete) usando archivos .rest.

Capturas de Pantalla:

Captura del login:

Appointments

LoginRegister

User Login

Correo electrónico

Enter your email


Contraseña

Enter your password

Iniciar sesión

Don't have an account? [Register here.](#)

Gestión de Citas Médicas

Fecha y Hora	Médico	Paciente	Motivo	Estado	Acciones
12/05/2025 21:09	Dra Ana Torres	Carlos Pérez	Dolor de cabeza	Cancelada	

Listado de usuarios:

List of Registered Users

Username	Email	Role
Dra Ana Torres	medico@clinica.com	Médico
Carlos Pérez	carlos@correo.com	Paciente
Administrador General	admin@clinica.com	Admin

Dashboard de paciente, aquí nos permite agendar citas y escribir el motivo:

Agendar Nueva Cita

Médico

Dra Ana Torres

Fecha y Hora

05/17/2025, 08:00 AM

Motivo

Dolor de cabeza recurrente.]

Agendar

Aquí observamos que se agendó la cita de manera correcta, también nos permite cancelar dicha cita.

Cita agendada exitosamente.

Gestión de Citas Médicas

+ Agendar nueva cita

Fecha y Hora	Médico	Paciente	Motivo	Estado	Acciones
17/05/2025 08:00	Dra Ana Torres	Carlos Pérez	Dolor de cabeza recurrente.	Agendada	⊗ Cancelar

Dashboard del médico:

Aquí se nos da la opción de marcar la cita como cancelada o realizada.

Gestión de Citas Médicas

Fecha y Hora	Médico	Paciente	Motivo	Estado	Acciones
17/05/2025 08:00	Dra Ana Torres	Carlos Pérez	Dolor de cabeza recurrente.	Agendada	⊗ Cancelar ⊕ Realizar

Al marcar “realizada” en el dashboard nos sale lo siguiente:

Gestión de Citas Médicas

Fecha y Hora	Médico	Paciente	Motivo	Estado	Acciones
17/05/2025 08:00	Dra Ana Torres	Carlos Pérez	Dolor de cabeza recurrente.	Realizada	

Pruebas con .rest:

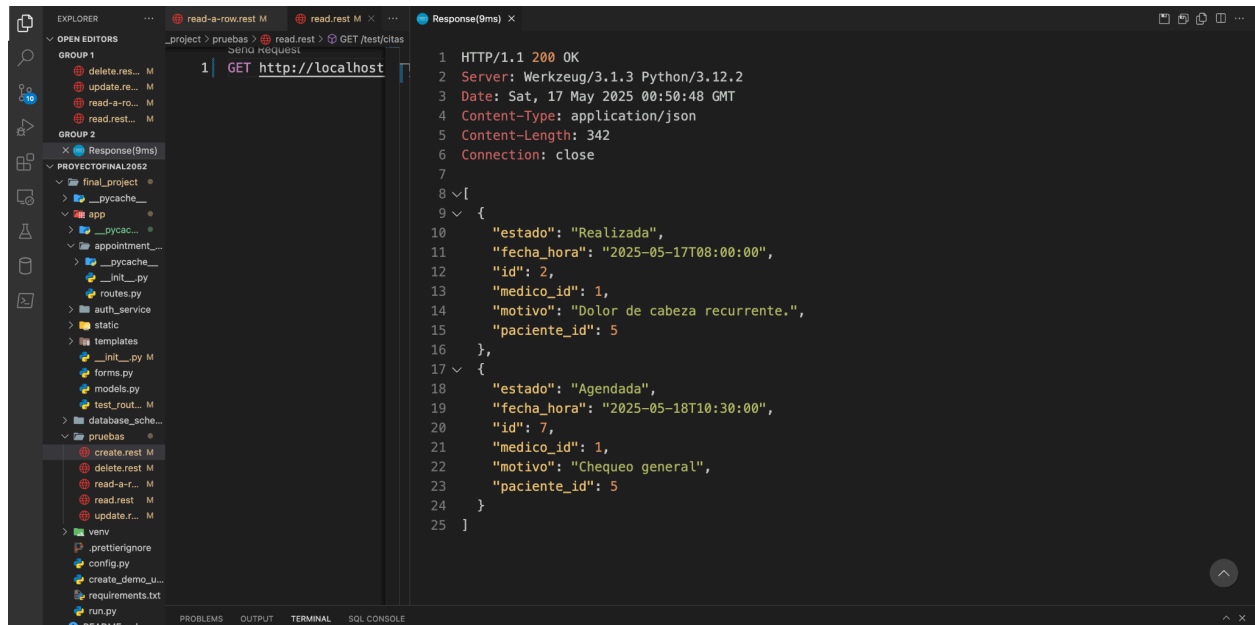
Crear cita:

```
final_project > pruebas > create.rest > POST /test/citas

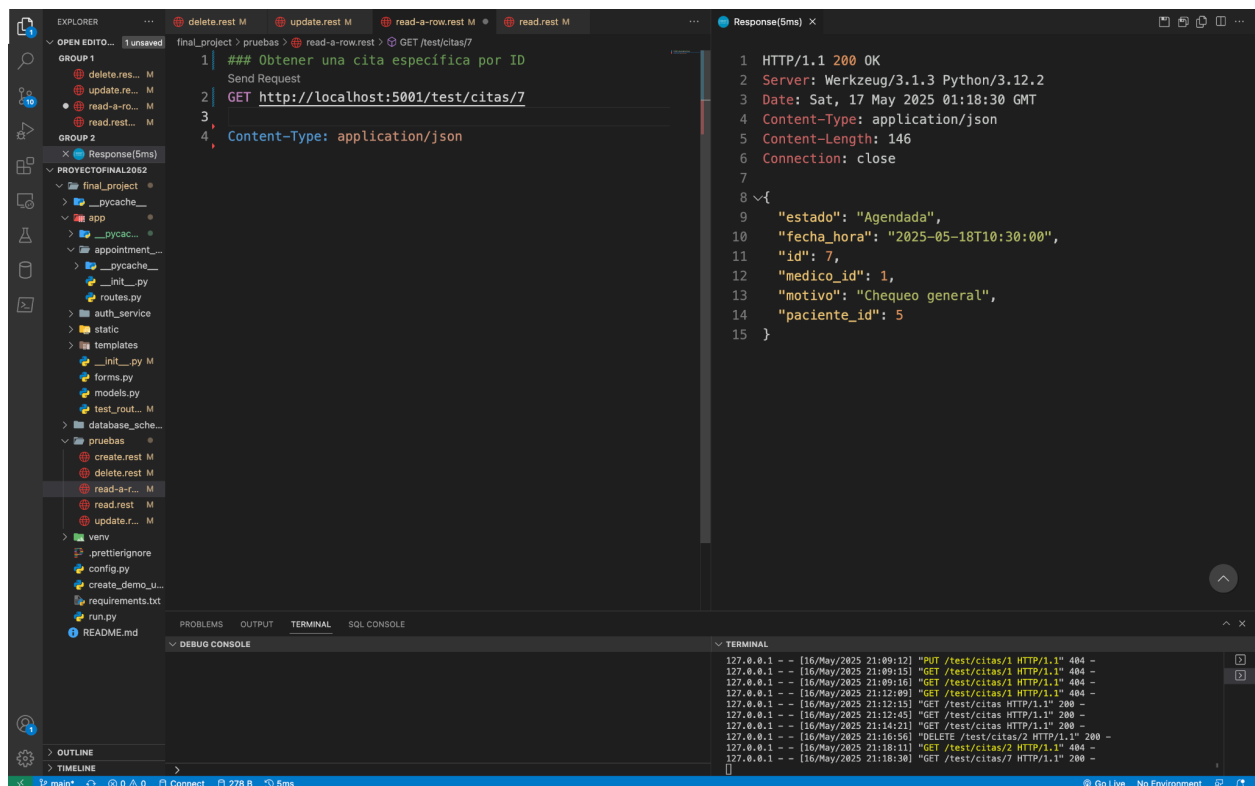
Send Request
1 POST http://localhost:5001/test/c
2 Content-Type: application/json
3
4 {
5   "fecha_hora": "2025-05-18T10:30
6   "motivo": "Chequeo general",
7   "medico_id": 1,
8   "paciente_id": 5
9 }

1 HTTP/1.1 200 OK
2 Server: Werkzeug/3.1.3 Python/3.12.
3 Date: Sat, 17 May 2025 00:35:27 GMT
4 Content-Type: application/json
5 Content-Length: 45
6 Connection: close
7
8 {
9   "message": "Cita creada correctam
   ente"
10 }
```

Obtener diferentes citas agendadas:



Obtener una cita en específico:



Borrar cita:

The screenshot shows a Visual Studio Code editor with a REST client configuration for deleting a citation. The Explorer sidebar on the left shows the project structure, including files like `delete.rest`, `update.rest`, `read-a-row.rest`, and `read.rest`. The main editor displays the REST client configuration for the `DELETE /test/citas/2` endpoint. The configuration includes a comment `## Eliminar una cita por ID`, a `Send Request` button, and the `DELETE` method with the URL `http://localhost:5001/test/citas/2` and `Content-Type: application/json`.

The response is shown in the `Response(19ms)` tab, displaying the following details:

- 1 HTTP/1.1 200 OK
- 2 Server: Werkzeug/3.1.3 Python/3.12.2
- 3 Date: Sat, 17 May 2025 01:16:56 GMT
- 4 Content-Type: application/json
- 5 Content-Length: 34
- 6 Connection: close
- 7
- 8 {
- 9 "message": "Cita eliminada"
- 10 }

The bottom of the screen shows the `TERMINAL` tab with a log of HTTP requests and responses, including the `DELETE /test/citas/2 HTTP/1.1" 200 -` request.