Bitácora semanal

Semana 1 (29/09 – 05/10)

Durante esta primera semana recibimos la letra del obligatorio y nos pusimos a leer todo detalladamente para entender bien qué nos estaban pidiendo. Analizamos todas las funcionalidades del sistema, las restricciones, los límites de uso de las salas y las reglas que debíamos cumplir sí o sí. Como el backend tenía que hacerse en Python y la base de datos debía ser MySQL, empezamos a pensar en cómo se iba a integrar todo más adelante. Con esa base, arrancamos a diseñar el primer modelo de datos, en base a las tablas que ya teníamos, decidir cómo se iban a relacionar, cuáles serían las claves primarias y foráneas, y cómo debíamos representar correctamente las reglas del sistema en la estructura de la base. Una vez que el modelo estuvo suficientemente sólido, pasamos a implementarlo. Creamos todas las tablas en SQL con sus tipos, relaciones y restricciones correspondientes.

Semana 2 (06/10 – 12/10)

En esta semana seguimos enfocándonos en la base de datos. Cuando la estructura quedó pronta, empezamos a cargar datos iniciales para poder realizar pruebas más adelante. También, implementamos todas las consultas obligatorias y las consultas extra que teníamos que agregar. A medida que las probábamos, ajustamos algunos detalles de la base y los datos para asegurarnos de que devolvieran bien los resultados. Fue una semana de mucha prueba y error, pero nos dejó la base bien afinada para poder avanzar con el backend sin problemas.

Semana 3 (13/10 – 19/10)

Esta semana la dedicamos a arrancar con el backend. Investigamos distintas opciones hasta que nos decidimos por Flask, dado que ya lo habíamos utilizado en uno de los retos de la carrera. Eso nos permitió avanzar más rápido sin tener que aprender algo desde cero. También, descubrimos el uso de los Blueprints, que nos dejaban organizar el backend por módulos, así que los incorporamos desde el inicio para mantener todo ordenado. También nos repartimos y comenzamos a implementar los endpoints principales.

Semana 4 (20/10 – 26/10)

En esta semana empezamos a investigar cómo dockerizar la base de datos, porque ninguna del equipo lo tenía completamente dominado. Los primeros días fueron bastante prueba y error, ya que no lográbamos conectar nada. Al mismo tiempo, arrancamos con el frontend y decidimos hacerlo en React Native, porque era lo que ya conocíamos y nos permitía armar una app más realista.

Semana 5 (27/10 – 02/11)

Después de varios intentos, esta semana por fin logramos terminar la dockerización de la base de datos. Dejamos el contenedor de MySQL funcionando de forma estable. Además, empezamos a informarnos sobre cómo conectar el backend y el frontend, cómo hacer las solicitudes, cómo usar los servicios, etc. Fue una semana más de investigación, pero clave para poder avanzar después. Una de las ventajas que descubrimos, es que Docker facilita la integración entre el backend (hecho con Flask) y la base de datos. Al tener la base corriendo en un contenedor, tenemos control total sobre el puerto, la contraseña, el usuario y la configuración general, lo que hace mucho más simple establecer la conexión desde Flask sin depender de instalaciones locales diferentes.

A continuación, se adjunta la estructura del contenedor, compuesta inicialmente por un entorno dedicado a la base de datos, y otro dedicado al framework de Flask.

Semana 6 (03/11 – 09/11)

Esta semana creamos los servicios del frontend, que son los encargados de comunicarse con la API. Con eso empezamos a integrar mejor cada parte de la app y a preparar todos los llamados necesarios al backend. Al mismo tiempo, seguimos avanzando con React Native para tener un frontend estable y lindo de ver, y también continuamos con algunas validaciones y ajustes en el backend para dejar todo más sólido.

Semana 7 (10/11 – 16/11)

En esta semana logramos conectar por completo el frontend con el backend, lo que nos permitió empezar a probar los flujos principales de la aplicación. Además, empezamos con el informe, ya que a esta altura podíamos documentar mejor cómo se había armado cada parte. También agregamos las restricciones finales en el backend para asegurarnos de que cada dato pasara por todas las validaciones necesarias antes de llegar a la base. Como último detalle para esta semana, creamos un proyecto en Postman para probar la app y asegurarnos de que todos los endpoints funcionaran correctamente con distintos escenarios.

Semana 8 (17/11 – 23/11)

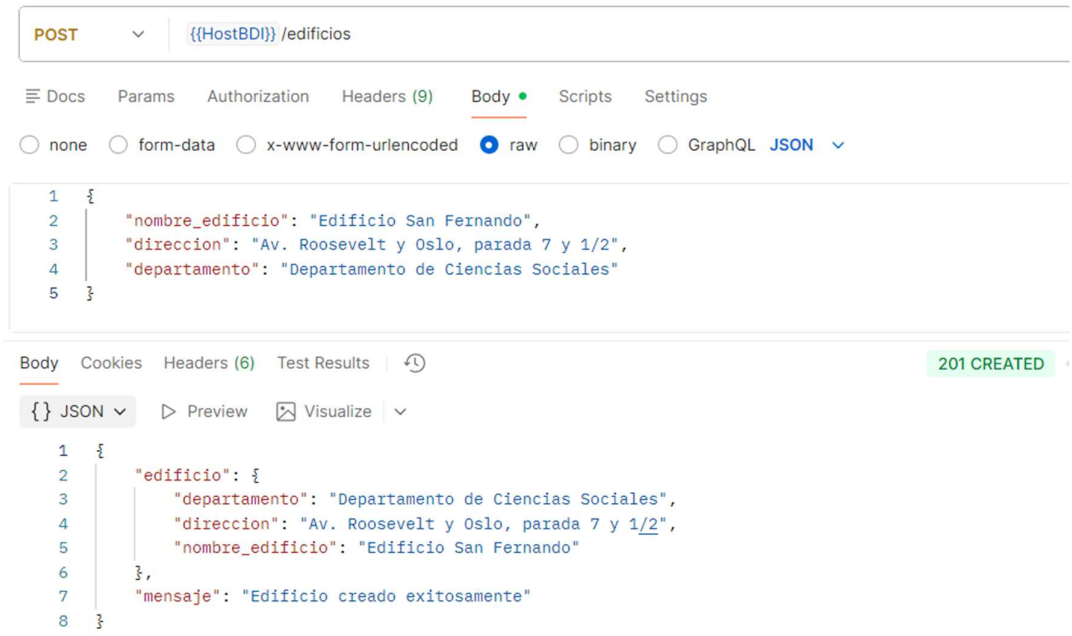
En la última semana cerramos todo lo que quedaba pendiente, hasta que nos dimos cuenta de que necesitábamos una persona que sea administrador para que pueda ver las estadísticas de las reservas y sanciones. Decidimos agregar el rol Admin como un usuario más para que sea el único con acceso a los reportes. También, el tema de la asistencia no sabíamos cómo implementarlo porque no teníamos bien definido si la responsabilidad era propia del usuario o de un administrador. Finalmente, decidimos que corra por parte del participante registrar si fue o no. En el caso de que no lo marque, por default no asistió.

Luego, terminamos el frontend y agregamos todas las validaciones necesarias para que coincidan con las del backend.

Con respecto al uso de Postman, hicimos todas las pruebas necesarias con los endpoints, asegurándonos que devuelvan los mensajes de tipo HTTP planificados. A continuación, se

adjuntan imágenes de la implementación correcta de algunos endpoints y sus resultados en la base de datos:

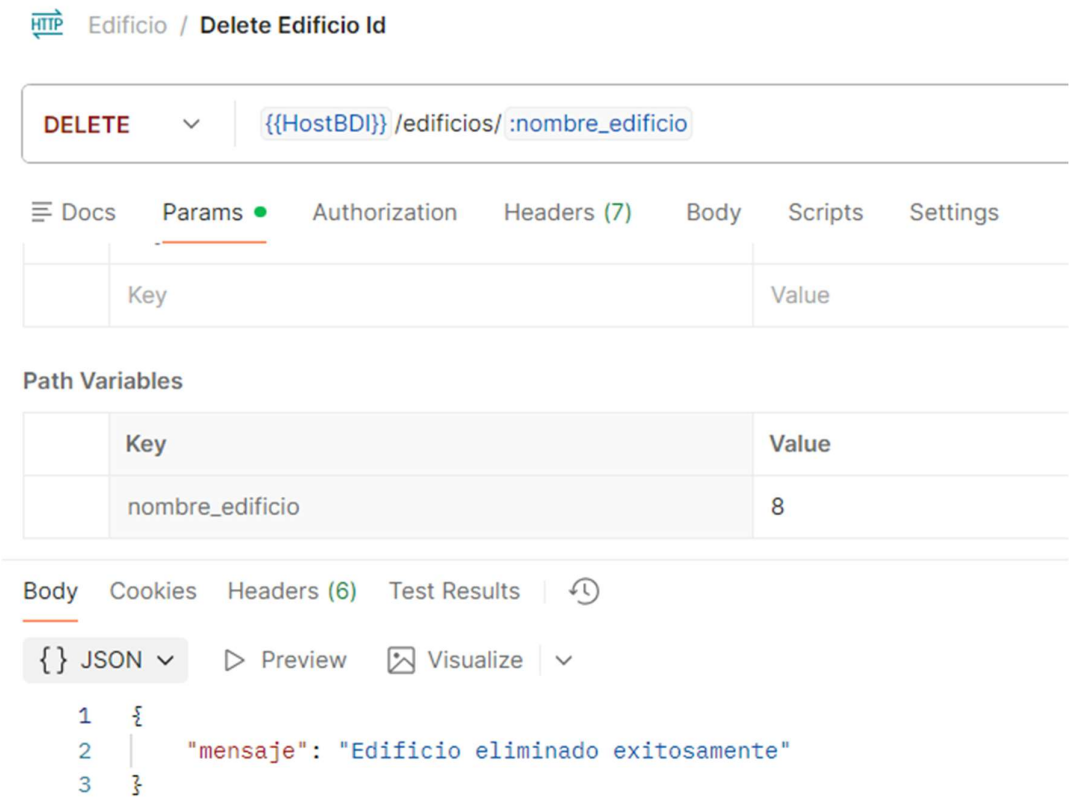
Creación de un edificio:



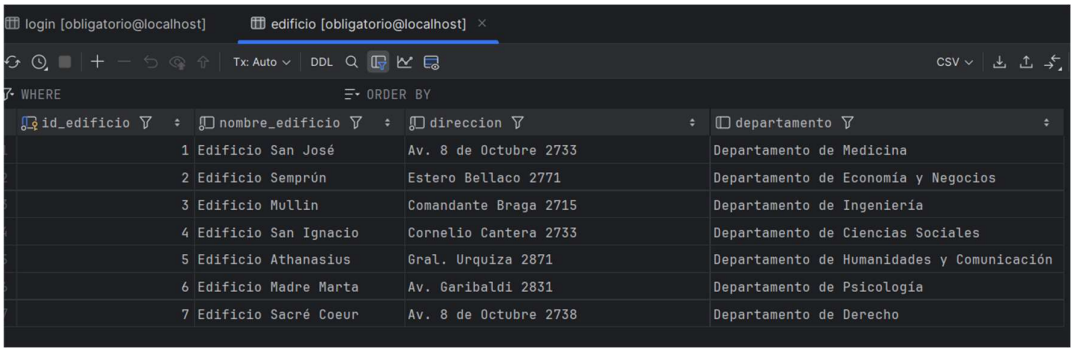
Resultado en base de datos:

	id_edificio	nombre_edificio	direccion	departamento
1	1	Edificio San José	Av. 8 de Octubre 2733	Departamento de Medicina
2	2	Edificio Semprún	Estero Bellaco 2771	Departamento de Economía y Negocios
3	3	Edificio Mullin	Comandante Braga 2715	Departamento de Ingeniería
4	4	Edificio San Ignacio	Cornelio Cantera 2733	Departamento de Ciencias Sociales
5	5	Edificio Athanasius	Gral. Urquiza 2871	Departamento de Humanidades y Comunicación
6	6	Edificio Madre Marta	Av. Garibaldi 2831	Departamento de Psicología
7	7	Edificio Sacré Coeur	Av. 8 de Octubre 2738	Departamento de Derecho
8	8	Edificio San Fernando	Av. Roosevelt y Oslo, parada 7 y 1/2	Departamento de Ciencias Sociales

Eliminación de un edificio:



Resultado en base de datos:



Además, terminamos de documentar todo el proceso, con las decisiones que tomamos, la estructura que elegimos y la bitácora la íbamos completando a medida que realizábamos las diferentes tareas, y redactamos el README final con todos los pasos explicados para ejecutar la aplicación. Con eso dejamos el proyecto completamente pronto para la entrega.