

Prompt para Replit - Sistema de Turnos Médicos

Quiero que revises y soluciones los siguientes problemas en mi proyecto de turnos médicos con React + Node + Supabase:

1. Reservas desde la página principal que no se guardan en Supabase (PROBLEMA PRINCIPAL):

- Cuando hago una reserva desde la página del paciente, aparece el mensaje "Reserva realizada con éxito" y me permite descargar el comprobante, pero la reserva no se guarda en la tabla appointments de Supabase.

- En cambio, si hago la reserva desde el panel de administrador, sí se guarda en la base de datos.

- Necesito que revises la función bookAppointment en el frontend (React/Next.js) y el flujo del backend (Node/Express si interviene) para asegurarte de que:

- * Se ejecute el insert en Supabase correctamente.

- * Se capture y muestre el error si la operación falla (ahora parece que ignora el error).

- * Solo se muestre el mensaje de éxito si el turno quedó realmente registrado en la base de datos.

2. Inputs poco legibles en el panel de administración (PC/Escritorio):

- Los campos de formularios (input) en el Panel de Administración se ven demasiado claros y no tienen buen contraste en pantallas de escritorio.

- Necesito que ajustes los estilos (CSS/Bootstrap/React) para que los inputs tengan bordes y texto más visibles (ejemplo: borde gris oscuro, fondo blanco definido, placeholder en gris medio).

- Que el diseño quede consistente con el resto de la interfaz.

3. Mejora en el panel de administración – Filtro por fecha:

- Agregar un filtro de búsqueda por fecha o rango de fechas en el Panel de Administración, para que el administrador pueda ver las reservas realizadas en un día específico o dentro de un rango seleccionado.

- El filtro debe integrarse con Supabase para traer únicamente las reservas correspondientes a esa fecha/rango.

- La interfaz debe mostrar solo los turnos filtrados, con opción de limpiar el filtro para volver a la vista general.

4. Problema de horarios y reservas (detallado en archivo .txt):

- Actualmente se generan intervalos de 30 minutos a partir de la tabla doctor_schedules y se consultan los turnos ocupados en la tabla appointments.

- En Supabase hay definida una restricción de unicidad (doctor_id, appointment_date, appointment_time).

- Revisar que la generación de slots y la verificación de disponibilidad se implementen correctamente para evitar que se muestre un turno ocupado como disponible.

5. Comandos para subir los cambios a GitHub desde la shell de Replit:

Después de aplicar los cambios, quiero que me muestres también los comandos necesarios para subir el código actualizado a mi repositorio de GitHub desde la terminal de Replit.

Los pasos deben ser:

- Verificar el estado del repo.

- Agregar todos los cambios.

- Hacer commit con un mensaje claro.
- Pushear al repositorio remoto en la rama principal (main o master).

Stack del proyecto:

- Frontend: React, TypeScript, Bootstrap, CSS
- Backend: Node.js, Express
- Autenticación: JWT
- Base de datos: Supabase
- Emails: Nodemailer

Entregame el código corregido y explícame los cambios que hiciste para que:

- Las reservas hechas desde la página principal se guarden en la base de datos Supabase de la misma forma que desde el administrador.
- Los inputs sean legibles en escritorio.
- El panel de administración tenga un filtro por fecha para gestionar las reservas.
- El sistema de slots funcione correctamente evitando turnos duplicados.
- Sepa qué comandos usar en la shell de Replit para subir los cambios a mi repositorio en GitHub.