

TRABAJO PRACTICO OBLIGATORIO

SITIO WEB PARA UN MEDICO

Desarrollar una aplicación web para un médico (especialidad a elección del grupo) que permita presentar información profesional, gestionar reservas de citas y administrar la agenda de turnos.

REQUERIMIENTOS GENERALES

- Cada grupo estará integrado por **dos alumnos como máximo**.
- La aplicación debe ser **responsive**.
- El **40%** de la nota se obtiene por el cumplimiento de las funcionalidades y la utilización de buenas prácticas que permitan la reutilización y optimicen la performance del sistema.
- El **40%** de la nota se obtiene de la usabilidad y experiencia de usuario de la aplicación desarrollada.
- El **20%** restante se obtiene de la **documentación correspondiente**.
- La aplicación deberá desarrollarse utilizando los siguientes lenguajes y librerías obligatorias: **HTML/CSS, React, JavaScript y NodeJS**.
- La base de datos a utilizar es **opción del grupo**, pudiendo escogerse entre **SQL** (MySQL, SQL Server, PostgreSQL) y **NoSQL** (MongoDB).

LANDING PAGE

- La página principal debe presentar al médico, su especialidad, formación, servicios ofrecidos y datos de contacto.
- Debe incluir un formulario o enlace que permita reservar una cita.
- Incluir un diseño responsive y profesional acorde a la temática médica.

RESERVA DE CITAS

- El formulario de reserva debe solicitar:
 - Nombre y Apellido
 - Nombre y apellido del paciente
 - Teléfono
 - Correo electrónico
 - Obra Social

- Debe incluir un calendario que muestre las citas disponibles durante las **dos próximas semanas**.
- Las fechas y horarios disponibles deben actualizarse dinámicamente según la ocupación.
- La obra social debe ser una lista de aquellas con las cuales el medico tiene convenio.

LOGIN PARA MÉDICO O SECRETARIA

- La landing page debe incluir un acceso para el médico o su secretaria mediante usuario y contraseña.
- **No** se implementará registro de usuarios. Las credenciales del administrador serán entregadas al médico una vez finalizado el desarrollo.

GESTIÓN DE CITAS (ÁREA ADMINISTRATIVA)

- Una vez logueados, el médico o su secretaria podrán:
 - Visualizar todas las citas solicitadas.
 - Confirmar una cita cambiando su estado de "Solicitada" a "Confirmada".

ADMINISTRACION DE OBRAS SOCIALES

- El sistema debe permitir al usuario crear, modificar o eliminar obras sociales.
- Estas serán las que se muestren en la funcionalidad de concertar cita.

NOTIFICACIONES

- Enviar notificación por correo electrónico al paciente cuando crea una nueva cita.
- Enviar notificación por correo electrónico al paciente cuando la cita pase de "Solicitada" a "Confirmada".

SEGURIDAD Y PRIVACIDAD

- Asegurar la protección de los datos de los pacientes y la privacidad de la información.
- Implementar buenas prácticas de seguridad, como el cifrado de contraseñas del administrador.

REQUERIMIENTOS NO FUNCIONALES

ESCALABILIDAD

- **Capacidad de manejar múltiples solicitudes de reserva simultáneamente.**

USABILIDAD

- **Interfaz clara y fácil de usar tanto para pacientes como para el médico o secretaria.**
- **Diseño adaptado a dispositivos móviles, tablets y ordenadores.**

RENDIMIENTO

- **Respuesta rápida en la carga del calendario y en la gestión de reservas.**

MANTENIBILIDAD

- **Código limpio, modular y documentado para facilitar futuras mejoras.**

COMPATIBILIDAD

- **Compatible con los principales navegadores web actuales.**

CRITERIOS DE EVALUACION

Categoría	Descripción	Ponderación
Funcionalidad	Cumplimiento de todos los requerimientos funcionales y no funcionales definidos en el enunciado. Incluye que el sistema permita reservar citas, mostrar disponibilidad, login de administrador y gestión de estado de las citas.	40%
Usabilidad y Experiencia de Usuario (UX)	Interfaz clara, intuitiva y responsive. Flujo de navegación coherente y adaptado a usuarios no técnicos. Diseño visual acorde a la temática médica.	25%
Calidad del Código y Buenas Prácticas	Uso correcto de HTML, CSS, React y NodeJS siguiendo buenas prácticas. Código modular, reutilizable y con comentarios claros. Manejo adecuado de estados en React y separación lógica en el backend.	15%
Seguridad y Validaciones	Protección de datos sensibles, cifrado de contraseñas, validación de formularios y manejo seguro de credenciales de administrador.	10%
Documentación	Documento que describa la arquitectura del sistema, instalación, configuración, credenciales de prueba y manual de usuario. Diagramas y justificación de la elección de tecnologías.	10%

