

TPI - 2025 - Programación III - Tema 1

Objetivo:

Desarrollar un sistema web que tenga administración de usuarios y perfiles. Debe gestionar reservas de distintos tipos de negocios, un ABM de los datos de dominio y un dashboard.

Especificaciones generales:

El sitio web debe tener 2 (dos) roles de usuario, uno ADMIN y el otro USUARIO.

Desde el rol ADMIN se debe poder crear los usuarios y cambiar la password.

Desde el rol de usuarios login y registro (si no ha iniciado sesión) y el menú con las opciones correspondientes y la opción de cerrar sesión).

Dado que no se dispone de un sistema de backend tipo API REST, este se emulará utilizando el servicio gratuito de terceros mockapi.io. Por lo tanto las entidades se deben definir y crear en [mockapi](https://mockapi.io).

El material de trabajo es HTML5, CSS3 y JavaScript.

No hay plantilla inicial con lo cual cada grupo tiene la libertad de utilizar alguna plantilla si lo desea o algún framework para el uso de los estilos (ej: Bootstrap).

El trabajo se debe realizar en grupos de 2 a 4 integrantes y el código debe estar subido a un repositorio Git para que podamos acceder al código fuente. Dicho repositorio debe ser público, no privado.

Criterios de Evaluación:

Maquetación: Armado correcto del sitio, uso de los TAGS.

Estilos: CSS prolijos y diseño responsivo (flex, grid, distintos dispositivos).

Codificación JavaScript: Manejo correcto de los eventos, asincronismo, arrow functions, uso de objetos y arrays.

Manejo del DOM: Lectura, escritura, sobre el árbol del DOM.

Seguridad: Correcto manejo de los roles y correcta asignación del menú de cada rol.

Dominio Tema 1: Sistema de Turnos Médicos

Contexto:

Una clínica pequeña necesita un sistema web para gestionar los turnos de sus pacientes.

Roles:

- **ADMIN:** gestiona médicos, pacientes y turnos.
- **USUARIO (Paciente):** puede registrarse, iniciar sesión, ver su perfil y solicitar turnos disponibles.

Entidades MockAPI:

- `users: { id, nombre, email, password, role }`
- `doctors: { id, nombre, especialidad, diasDisponibles }`
- `appointments: { id, patientId, doctorId, fecha, hora, estado }`

Requisitos principales:

- Formulario de login/registro con validación.
- Vista de admin con listado de médicos y turnos.
- Vista de usuario con selección de médico y reserva de turno.
- Posibilidad de cancelar turno (usuario) o confirmarlo (admin).
- Un dashboard con un gráfico con el estado de los turnos.