**Sistema de Control de Acceso para Piscinas**

**Enunciado (Reto)**

El Complejo de Piscinas de la Unidad Deportiva Atanasio Girardot requiere un control de entradas, permanencia y salida de los usuarios. Se requiere un sistema básico de información para el registro de usuarios que acceden a las instalaciones, con la posibilidad de modificación, eliminación y consulta de dichos registros. En particular, los requerimientos plantean la construcción de un sistema de información básico FrontEnd/BackEnd que implemente un Servidor NodeJS y una interfaz Web de Interacción a través de llamadas REST y respuestas de códigos HTTP que realice operaciones CRUD sobre una base de datos de usuarios de las instalaciones.

**Estructura del Proyecto**



**Instalación y Ejecución**

**1.- Crear la base de datos:**

* **Base de Datos PostgreSQL -> nombre: “piscinas”**
* ***server/migrations/001-create-usuarios.sql***
* ***En el IDE pgAdmin4, ejecute la instrucción de creación de la tabla usuarios***

**CREATE TABLE usuarios (**

**id SERIAL PRIMARY KEY,**

**nombre VARCHAR(100) NOT NULL,**

**documento VARCHAR(20) UNIQUE NOT NULL,**

**fecha\_ingreso TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,**

**fecha\_salida TIMESTAMP,**

**estado VARCHAR(20) DEFAULT 'DENTRO'**

**);**

**CREATE INDEX idx\_usuarios\_estado ON usuarios(estado);**

**2.- Instalar dependencias:**

**npm init -y**

**npm install express pg cors**

**3.- Crear las carpetas y cargar los archivos (estructura) del proyecto:**

**4.- Iniciar el servidor:**

**node server/server.js**

**5.- Arrancar la página Web**

* Abrir el archivo client/index.html en un navegador web.

**6.- Características del Sistema**

* Interfaz Responsiva: Diseño adaptable a diferentes tamaños de pantalla con colores pastel.
* Operaciones CRUD:
  + CREATE (POST): Registrar nuevo usuario en las piscinas
  + READ (GET): Listar todos los usuarios registrados
  + UPDATE (PUT): Modificar datos de un usuario existente
  + DELETE (DELETE): Eliminar un usuario del sistema
* Feedback Visual: Muestra claramente el código de estado HTTP y la respuesta del servidor.
* Validaciones: Manejo básico de errores y validaciones en el cliente y servidor.