

Manual de Usuario - CRUD de Gestión Escolar de Codesa

1. Introducción

1.1 Propósito

Este manual proporciona instrucciones detalladas para el uso del sistema CRUD Escolar desarrollado para la prueba técnica de Codesa. El sistema permite gestionar estudiantes, profesores y consultar información de personas.

1.2 Alcance del producto

El sistema está compuesto por:

- Un frontend Angular para interacción visual del usuario con formularios reactivos.
- Un backend Spring Boot que expone servicios RESTful.
- Módulos específicos para Estudiantes, Profesores y Personas.

1.3 Definiciones, acrónimos y abreviaciones

CRUD: Create, Read, Update, Delete (Crear, Leer, Actualizar, Eliminar)

MVC: Modelo Vista Controlador

API: Application Programming Interface

DTO: Data Transfer Object

REST: Representation State Transfer

JPA: Java Persistence API

JSON: JavaScript Object Notation

2. Información general del usuario

Este sistema está diseñado para ser utilizado por personal administrativo o académico encargado de gestionar los registros escolares de estudiantes y profesores.

3. Requisitos del sistema

3.1 Requisitos del cliente

- Navegador web moderno (Chrome, Firefox)
- Node.js 18+ (para frontend)
- Java 17 (para backend)

3.2 Instalación del sistema

- Clonar el repositorio del frontend (<https://github.com/santiagoafl/frontend-codesa>) y backend (<https://github.com/santiagoafl/backend-codesa>) alojados en GitHub.
- Ejecutar *npm install* y *ng serve* en el frontend.
- Ejecutar *mvn spring-boot:run* en el backend.

4. Guía de uso

4.1 Inicio de la aplicación

- La base de datos del backend se inicia en <http://localhost:8080/h2-console>
- El frontend se abre en <http://localhost:4200>

4.2 Gestión de Estudiantes

- Accede al módulo Estudiantes
- Puedes crear, editar, eliminar y listar estudiantes
- Requiere: nombre, apellido, email, teléfono, grado, matrícula, fecha de nacimiento

4.3 Gestión de Profesores

- Accede al módulo Profesores
- Puedes crear, editar, eliminar y listar profesores
- Requiere: nombre, apellido, email, teléfono, especialidad, fechas

4.4 Vista de Personas

- Accede al módulo Personas
- Lista combinada de estudiantes y profesores
- Solo lectura

4.5 Validaciones del sistema

- Todos los campos obligatorios deben completarse
- No se aceptan fechas futuras
- Teléfono es solo numérico
- Errores con SweetAlert2

5. Funcionalidades avanzadas

- Manejador global de errores HTTP
- Interfaz responsive con Angular Material
- Interceptor de errores visuales

6. Soporte

Para soporte técnico, contactar con el desarrollador del sistema o crear un issue en GitHub.

8. Anexos

- Script *carga_estudiantes_profesores.sql*
- Imagen con esquema de la base datos
- Postman Collection usado para verificar el funcionamiento del Backend