

Programación Web

Buscar



Página web - EduConecta

Equipo 1 - Plan B

Integrantes:

- Adauto Sedano Alexander
- Perez Cerron Dayana

Introducción



EduConecta es una aplicación web desarrollada para digitalizar y simplificar el proceso de matrícula escolar en la provincia de Huancayo. La solución permite a las familias consultar vacantes disponibles en colegios, completar el formulario de matrícula con datos del estudiante y apoderados, adjuntar la documentación requerida y registrar la matrícula en una base de datos segura administrada en MySQL. La aplicación es pública, no requiere inicio de sesión y está pensada para usuarios con poca experiencia digital.



Problemática



En Huancayo el trámite de matrícula es mayoritariamente presencial, lo que provoca largas filas, falta de información centralizada sobre vacantes y pérdida de tiempo para las familias. Esta situación afecta especialmente a quienes trabajan, a personas en zonas alejadas y a quienes tienen dificultades de movilidad.



Objetivos



El objetivo general es facilitar y acelerar la matrícula escolar mediante una plataforma accesible. Objetivos específicos: centralizar información de vacantes, permitir matrícula digital sin login, reducir trámites presenciales y garantizar la integridad de los registros en la base de datos.



Beneficiarios



Beneficiarios directos: padres, apoderados y estudiantes de Huancaayo. **Beneficiarios indirectos:** instituciones educativas, personal administrativo y autoridades locales, que obtienen datos organizados y menos sobrecarga operativa.



Marco Teórico

Buscar



EduConecta contribuye al acceso equitativo a la educación al ofrecer información transparente sobre vacantes escolares y permitir la matrícula remota, reduciendo tiempos, costos y desigualdades para las familias de Huancayo. El desarrollo del sistema respetó principios éticos como la privacidad y la minimización de datos, solicitando solo la información necesaria sin pedir celular ni correo, además de utilizar tecnologías y recursos que cumplen con licencias de software libre. Finalmente, el proyecto garantiza sostenibilidad al emplear herramientas estables y de bajo costo como Angular, Node y MySQL, complementadas con documentación y capacitación básica que permite a las instituciones administrar la base de datos de forma segura mediante DBeaver.

Arquitectura de la Aplicación Web



- Frontend (Angular): componentes HTML/TS, formularios, services HttpClient.

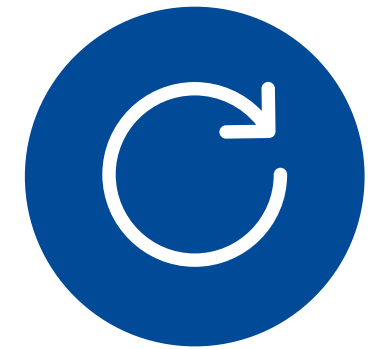


- Backend (Node + Express): endpoints REST mínimos para listar y registrar matrículas.



- Base de datos (MySQL, gestionado con DBeaver): tablas relacionales y procedimientos almacenados.

Medidas de seguridad implementadas



1

- Validación en frontend: campos obligatorios (DNI 8 dígitos, nombres, género), documentos válidos.

2

- Validación en backend: revalidación de formatos y estructura.

3

- Consultas parametrizadas y uso de procedimientos almacenados (CREATE PROCEDURE) para todas las inserciones y selecciones, evitando inyección SQL.

4

- Rutas limitadas: solo POST /registrar y GET /matriculas.

5

- Manejo controlado de errores: mensajes claros, sin exponer información técnica.

DISEÑO DE BASE DE DATOS



Descripción de las Tablas Principales



- **schools**

Almacena los datos de los colegios, incluyendo nombre, dirección, teléfono y estado. Sirve como base para la asignación de matrículas y la gestión de vacantes.

- **people**

Registra información personal tanto de estudiantes como de apoderados, centralizando los datos en un solo punto para evitar duplicidades.

- **parents**

Define la relación entre estudiante y apoderado. Permite registrar múltiples responsables para un mismo estudiante.

- **students**

Identifica a cada estudiante dentro del sistema, vinculándolo con su información personal y las futuras matrículas.



- **grades**

Contiene los grados escolares, permitiendo seleccionar el nivel al cual será matriculado el estudiante.

- **grades_in_schools**

Gestiona las vacantes disponibles por grado y por institución educativa, siendo fundamental para validar la disponibilidad antes de registrar una matrícula.

- **enrollments**

Registra la matrícula del estudiante, relacionando colegio, grado y persona.

- **documents**

Almacena la referencia a los archivos adjuntos enviados durante el proceso de matriculación.





Relaciones entre Tablas

- people → students: relación 1 a 1 para estudiantes.
- people → parents → people: relación persona-apoderado mediante tabla intermedia.
- students → enrollments → schools & grades: conecta estudiante, colegio y grado dentro de la matrícula.
- grades_in_schools → schools / grades: determina cuántas vacantes tiene cada grado en cada institución.
- documents → enrollments: registra los archivos enviados por estudiante.



Procedimientos Almacenados

Tipos de procedimientos:

- sp_sel_* → obtienen listas, filtros y búsquedas.
- sp_ins_* → registran personas, estudiantes, padres, matrículas y documentos.
- sp_upd_* → usados para administración interna.

Ventajas:

- Evitan inyección SQL.
- No se permite acceso directo a las tablas desde el backend.
- Centralizan y ordenan la lógica del sistema.
- Retornan el ID de los registros creados para continuar el flujo.





Lógica de Matrícula

El proceso de matrícula está controlado por la base de datos mediante los procedimientos:

1. Se consulta la tabla `grades_in_schools` para verificar vacantes.
2. Si hay disponibilidad, se registran los datos personales del estudiante usando `sp_ins_people`.
3. Luego se crea el estudiante en la tabla `students`.
4. Se registran los apoderados en `parents`.
5. Se inserta la matrícula en la tabla `enrollments`.
6. Finalmente, se adjuntan documentos en la tabla `documents`.





Gestión de Vacantes

La disponibilidad se registra en la tabla `grades_in_schools`, que contiene cuántos cupos tiene cada colegio según grado. Antes de registrar una matrícula, la aplicación consulta esta tabla.

La actualización de vacantes es controlada por personal autorizado desde DBeaver, asegurando precisión y trazabilidad.



Documentos de Matrícula

La tabla `documents` guarda el archivo subido por el usuario.

Se almacena la ruta en el servidor y se mantiene protegido contra accesos no autorizados.

El procedimiento `sp_ins_documents` asegura que el documento corresponda a una matrícula existente.





Demonstración de EduConecta

