Proyecto: Aulas Resilientes

Sistema Educativo Offline/Online para Zonas Rurales



La región es una zona con:

- Frecuentes cortes de energía e internet.
- Flora y fauna autóctona (caldenes, ñandúes, vizcachas) que inspiran el diseño.
- Comunidad educativa unida (docentes, padres y alumnos colaborativos).

Objetivo: Crear una plataforma económica, offline-first y escalable que funcione en cualquier dispositivo.

📌 Características Clave

1. Para Docentes

- **Crear tareas offline** y sincronizarlas luego.
- **Chat colaborativo** por materia/grado.
- Registrar asistencia sin internet.

2. Para Alumnos

- Se Acceder a materiales (videos, PDFs, ejercicios) sin conexión.
- **Enviar tareas** desde su celular (fotos, audio, texto).

3. Para Administrativos

- **Generar reportes** en PDF automáticos.
- Sincronizar datos entre escuelas vía USB/red local.

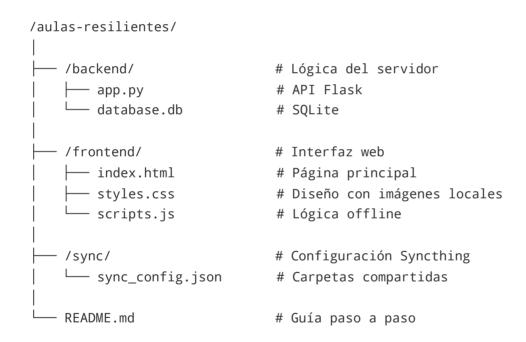
4. Para Padres (Opcional)

- Ver calificaciones guardadas offline.
- A Recibir notificaciones vía SMS/Telegram.

X Tecnología Utilizada

Componente	Herramienta	Costo (ARS)
Frontend	PWA (HTML5 + JavaScript)	\$0
Backend	Python + Flask + SQLite	\$0
Sincronización	Syncthing (red local)	\$0
Videollamadas	Jitsi Meet (autoalojado)	\$0
Hosting	PC vieja o Raspberry Pi	\$80.000

Estructura del Proyecto



Diseño con Identidad Local

Paleta de colores:

Tierra Pasto Cielo

Imágenes incluidas:

- Fondo de login: Atardecer en la llanura.
- Íconos: Árboles autóctonos, fauna local, cultivos.

Código Clave

1. Backend (Python + Flask)

```
app.py
 from flask import Flask, request, jsonify
 import sqlite3
 from datetime import datetime
 app = Flask(__name___)
 # Base de datos local
 def get_db():
     return sqlite3.connect('database.db')
 # Ruta para subir tareas offline
 @app.route('/upload_task', methods=['POST'])
 def upload_task():
     data = request.json
     conn = get_db()
     conn.execute(
         "INSERT INTO tasks (student_id, content) VALUES (?, ?)",
         (data['student_id'], data['content'])
     )
     conn.commit()
     return jsonify({"status": "success"})
 if __name__ == '__main__':
     app.run(host='0.0.0.0', port=5000)
```

2. Frontend (PWA - JavaScript)

```
// Guardar tareas offline
function saveTask(task) {
   if (!navigator.onLine) {
      localStorage.setItem('pending_tasks', JSON.stringify(task));
      alert("Tarea guardada para sincronizar luego.");
```

```
} else {
     sendToServer(task);
}

// Sincronizar al reconectar
window.addEventListener('online', () => {
     const tasks = JSON.parse(localStorage.getItem('pending_tasks'));
     if (tasks) sendToServer(tasks);
});
```

III Presupuesto Detallado

Item	Costo (ARS)	Notas
Raspberry Pi 4	\$120.000	Servidor local.
USBs 32GB (x10)	\$30.000	Distribución de contenido.
Cartelería	\$15.000	Afiches tutoriales.
Total	\$165.000	(≈ USD 150 al cambio informal).

Implementación en 3 Pasos

- 1. Semana 1: Instalar el servidor en una PC vieja.
- 2. **Semana 2**: Capacitar a docentes (2 talleres prácticos).
- 3. Mes 1: Prueba piloto en una escuela local.

M Beneficios

- ✓ 100% offline: Funciona sin internet.
- ✓ Económico: Menos de \$200.000 ARS iniciales.
- ✓ Escalable: Se puede replicar en otras escuelas rurales.



```
# En una escuela sin internet:
docente = Docente(nombre="María")
docente.subir_tarea_offline("Matemática - Ejercicios página 45")
alumno = Alumno(nombre="Juan")
alumno.enviar_tarea(foto="ejercicios_resueltos.jpg")
```

Acceso para Padres (Opcional)

Si hay internet móvil:

- Web app: http://aulasresilientes.local.edu.ar
- Notificaciones: SMS gratis con Twilio o Telegram Bot.

¿Listos para empezar?

- 📥 <u>Descargar código completo</u>
- 📞 Contacto: equipo@aulasresilientes.ar

"La educación no se detiene, incluso cuando el internet sí."

Interfaz inspirada en el entorno local.