

Portal Web con Importador Inteligente - Opción 3

Solución Completa para Carga Masiva de Terceros

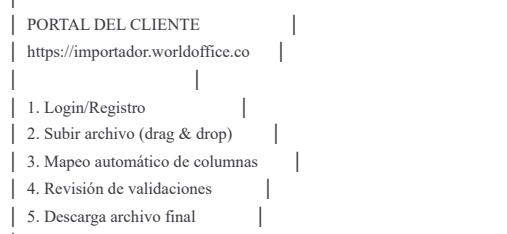
⌚ CONCEPTO GENERAL

Un portal web donde los clientes pueden subir archivos de terceros en CUALQUIER formato (Excel caótico, CSV exportado de otro sistema, etc.) y el sistema automáticamente:

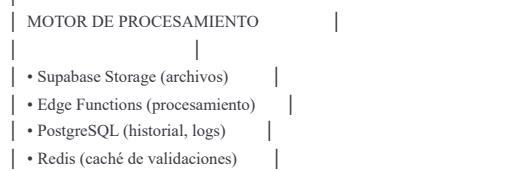
1. Detecta las columnas y las mapea a los campos de WorldOffice
2. Valida y corrige datos
3. Enriquece información faltante
4. Genera Excel perfecto para importar

🏗 ARQUITECTURA TÉCNICA

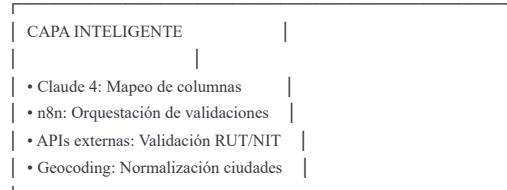
Frontend (React + Tailwind)



Backend (Supabase + Edge Functions)



Inteligencia (Claude API + n8n)



⌚ FLUJO COMPLETO PASO A PASO

PASO 1: Cliente sube archivo

```
javascript
// El cliente arrastra cualquier archivo
Formatos aceptados:
- Excel (.xlsx, .xls)
- CSV (.csv, .txt)
- Google Sheets (link)
- JSON exportado de otro sistema

Tamaño máximo: 50MB (≈ 100,000 registros)
```

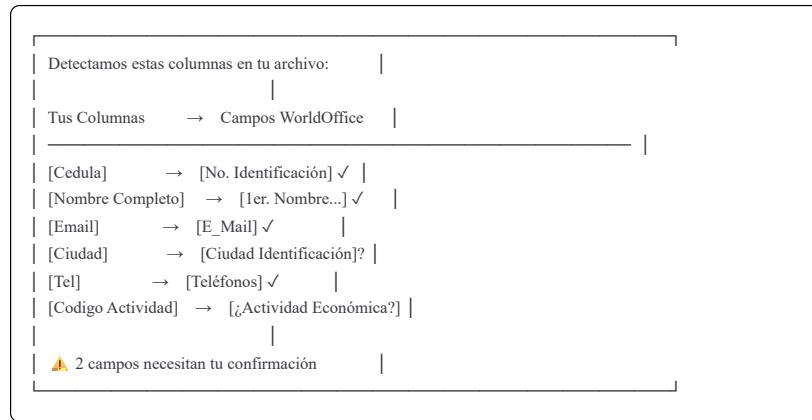
Lo que sucede en el backend:

1. Archivo sube a Supabase Storage

2. Se extrae metadata: nombre, tamaño, formato
 3. Se parsea y se obtiene preview de primeras 5 filas
 4. Se crea registro en tabla `imports` con estado "uploaded"
-

PASO 2: Mapeo Inteligente de Columnas

Interfaz Visual:



Motor IA para mapeo (Claude API):

```
javascript
// Prompt a Claude
const prompt = `

Dado este header de Excel:
${columnasCliente}

Y estos campos obligatorios de WorldOffice:
${camposWorldOffice}

Mapea automáticamente las columnas que coinciden.
Responde en JSON con confianza de 0-100 para cada mapeo.
`;

// Respuesta de Claude
{
  "mapeos": [
    { "origen": "Cedula", "destino": "No. Identificación", "confianza": 95 },
    { "origen": "Nombre Completo", "destino": "1er. Nombre o Razón Social", "confianza": 85 },
    { "origen": "Ciudad", "destino": "Ciudad Identificación", "confianza": 60 }
  ],
  "dudosos": ["Codigo Actividad"], // Confianza < 70
  "no_mapeados": ["Personalizado 1", "Personalizado 2", ...]
}
```

El cliente puede:

- Aceptar mapeo sugerido
 - Ajustar manualmente los dudosos
 - Ignorar columnas que no necesita
-

PASO 3: Validación Inteligente en Tiempo Real

Motor de Validación Multi-capas:

Capa 1: Validaciones Estructurales

```
javascript
Validaciones automáticas:
✓ Tipo Identificación: Debe estar en lista ["NIT", "CC", "CE", "PP"]
✓ No. Identificación: Solo números, dígito de verificación correcto
✓ Email: Formato válido con @@
✓ Teléfonos: Formato colombiano (10 dígitos)
✓ Fechas: Formato ISO o DD/MM/YYYY
```

Capa 2: Validaciones Contextuales

javascript

Validaciones inteligentes:

- ✓ Si Tipo ID = "NIT" → debe tener dígito verificación
- ✓ Si Tipo ID = "CC" → debe tener entre 6-10 dígitos
- ✓ Si "Persona Natural" → requiere nombres y apellidos
- ✓ Si "Persona Jurídica" → solo requiere Razón Social
- ✓ Ciudad debe existir en catálogo de Colombia (1,122 ciudades)

Capa 3: Validaciones Externas (Opcional)

javascript

APIs de validación:

- ✓ **DIAN API:** Valida que NIT exista realmente
- ✓ Google Geocoding: Normaliza direcciones y ciudades
- ✓ **NIT Validation:** Calcula dígito de verificación
- ✓ Nomenclatura Direcciones: Valida formato "Calle 1 # 2-3"

Dashboard de Validación:

RESULTADOS DE VALIDACIÓN

Total registros: 1,247

✓ Válidos: 1,089 (87%)

⚠ Advertencias: 142 (11%)

✗ Errores críticos: 16 (2%)

[Ver errores críticos] [Ver advertencias]

[Descargar válidos] [Exportar reporte]

PASO 4: Corrección Guiada de Errores

Vista de Errores por Registro:

Fila 23: Juan Pérez - NIT 900123456

✗ Dígito de verificación incorrecto

Ingresaste: 900123456-7

Debería ser: 900123456-2

[Corregir automáticamente]

⚠ Email sin formato válido

Ingresaste: juanperez@gmail

Sugerencia: juanperez@gmail.com

[Aplicar sugerencia]

⚠ Ciudad no encontrada: "Bogotá DC"

Coincidencias cercanas:

• Bogotá D.C. ← Recomendado

• Bogotá

[Seleccionar]

Correcciones Masivas:

CORRECCIONES SUGERIDAS EN BLOQUE

Se detectaron 45 registros con:

"Bogotá DC" → Cambiar a "Bogotá D.C."

[Aplicar a todos] [Revisar uno por uno]

PASO 5: Enriquecimiento Automático

Auto-completado Inteligente:

Regla 1: Detección de Tipo de Persona

```
javascript

if(tipoID === "NIT") {
    // Es persona jurídica
    campo_razonsocial = primer_nombre;
    primer_nombre = "";
    apellidos = "";
} else if(tipoID === "CC" || tipoID === "CE") {
    // Es persona natural
    // Dividir nombre completo en componentes
    dividirNombreCompleto(nombre_completo);
}
```

Regla 2: Clasificación Tributaria por defecto

```
javascript

if(tipoID === "NIT" && nit.startsWith("9")) {
    clasificacion_dian = "49 - Otros contribuyentes";
    responsabilidades = ["O-13 - Gran contribuyente"];
} else {
    clasificacion_dian = "05 - Persona natural";
    responsabilidades = ["R-99-PN - No responsable de IVA"];
}
```

Regla 3: Valores por Defecto

```
javascript

Campos con defaults inteligentes:
✓ Propiedad Activa: "SI" (99% de los casos)
✓ Activo: "Activo"
✓ Plazo: 30 (días estándar)
✓ Aplica Retelca: "NO" (si no se especifica)
✓ Maneja Cupo Crédito: "NO"
✓ Fecha Creación: Fecha actual
```

PASO 6: Generación de Excel Final

Estructura de salida:

```
📁 Paquete de descarga (ZIP):
├── 📄 terceros_importar_worldoffice.xlsx (formato exacto 62 columnas)
├── 📄 reporte_validacion.pdf (errores encontrados y corregidos)
├── 📄 log_transformaciones.txt (qué se cambió en cada registro)
└── 📄 registros_rechazados.xlsx (si hubo errores no corregibles)
```

Excel generado incluye:

- Hoja "migrar clientes proveedores" con 62 columnas exactas
- Hoja "Listas01" con catálogos de referencia
- Hoja "sucursales Clientes" si aplica
- Formato y validaciones preservadas

💡 CARACTERÍSTICAS AVANZADAS

1. Plantillas Guardadas

El cliente puede guardar su mapeo:
"Exportación desde SAP" → Plantilla guardada

Próxima vez que suba archivo similar:
Aplicación automática del mapeo previo

2. Procesamiento por Lotes

Para archivos > 10,000 registros:

- Procesamiento asíncrono
- Email cuando termine
- Progress bar en tiempo real

3. Historial de Importaciones

Dashboard del cliente:

Fecha	Archivo	Registros	
2024-11-20	clientes.xlsx	1,247 ✓	
2024-11-15	proveedores.csv	856 ✓	
2024-11-10	terceros.xls	3,421 ⚠	

4. API para Integraciones

javascript

```
// Otros sistemas pueden usar el importador
POST /api/v1/import
{
  "file_url": "https://...",
  "mapping": {...},
  "webhook_url": "https://callback..."
}
```

🛠 STACK TECNOLÓGICO PROPUESTO

Frontend

- React 18 + TypeScript
- Tailwind CSS + shadcn/ui
- React Query (estado y caché)
- React Dropzone (subida de archivos)
- AG Grid (tabla de datos)
- Recharts (gráficos de validación)

Backend

- Supabase (base de datos + storage + auth)
- Edge Functions (procesamiento)
- Redis (caché de validaciones)
- PostgreSQL (datos estructurados)

Procesamiento

- n8n (orquestación de validaciones)
- Claude 4 API (mapeo inteligente)
- Papa Parse (parsing de CSV)
- ExcelJS (generación de XLSX)

Validaciones Externas

- DIAN API (validación RUT/NIT)
- Google Geocoding API (normalización direcciones)
- SendGrid (emails de notificación)

📊 MÉTRICAS DE RENDIMIENTO ESPERADAS

Archivo con 1,000 registros:

- Subida: 3-5 segundos
- Mapeo IA: 5-8 segundos
- Validación: 15-20 segundos
- Correcciones: Manual (según errores)
- Generación Excel: 5-10 segundos

TOTAL: ~1-2 minutos (sin correcciones manuales)

Archivo con 10,000 registros:

- └─ Procesamiento: 2-3 minutos (asíncrono)
- └─ Notificación por email cuando termine

💰 COSTOS ESTIMADOS DE OPERACIÓN

Por importación de 1,000 registros:

- Claude API (mapeo): \$0.015
- Supabase Storage: \$0.001
- Supabase DB: \$0.002
- APIs validación: \$0.010 (si se usan todas)
- n8n ejecuciones: \$0.000 (self-hosted)

TOTAL: ~\$0.03 USD por importación de 1,000 registros

Infraestructura mensual:

- Supabase Pro: \$25/mes
- Vercel Pro (hosting): \$20/mes
- Claude API: Variable según uso
- Redis Cloud: \$10/mes

TOTAL: ~\$55-75 USD/mes + uso

⌚ TIEMPO DE DESARROLLO

Fase 1: MVP (2 semanas)

- ✓ Subida de archivos
- ✓ Mapeo manual de columnas
- ✓ Validaciones básicas (estructura)
- ✓ Generación de Excel
- ✓ Login/Auth básico

Fase 2: IA y Validaciones (1 semana)

- ✓ Integración Claude API (mapeo automático)
- ✓ Validaciones contextuales
- ✓ Enriquecimiento de datos
- ✓ Corrección guiada

Fase 3: Optimizaciones (1 semana)

- ✓ Procesamiento asíncrono
- ✓ APIs externas (DIAN, Geocoding)
- ✓ Plantillas guardadas
- ✓ Historial de importaciones
- ✓ Dashboard administrativo

TOTAL: 4 semanas (1 mes)

🎁 VALOR AGREGADO PARA WORLD OFFICE

Para los Clientes:

- Reduce tiempo de carga de 2-3 días a 30 minutos
- Elimina 80% de errores en importación
- No necesitan aprender formato complejo
- Pueden subir desde cualquier sistema

Para WorldOffice:

- Reduce tickets de soporte en 60%
- Acelera onboarding de nuevos clientes
- Datos más limpios en el sistema
- Dashboard de métricas de importaciones
- Ventaja competitiva vs otros softwares

ROI Estimado:

Antes:

- Promedio 3 horas de soporte por cliente con problemas de carga
- 40% de clientes tienen problemas
- $100 \text{ clientes/mes} \times 40\% \times 3 \text{ hrs} \times \$15/\text{hr} = \$1,800/\text{mes}$ en soporte

Después:

- Problemas reducidos a 10%
- $100 \text{ clientes/mes} \times 10\% \times 0.5 \text{ hrs} \times \$15/\text{hr} = \$75/\text{mes}$

AHORRO: \$1,725/mes

Recuperación inversión: 1.5 meses

PRÓXIMOS PASOS

Si decidimos avanzar con esta opción, el proceso sería:

Semana 1:

- Diseño de interfaces (Figma)
- Setup de infraestructura (Supabase + Vercel)
- Desarrollo de subida de archivos + parsing

Semana 2:

- Sistema de mapeo manual
- Validaciones básicas
- Generación de Excel

Semana 3:

- Integración Claude API
- Validaciones inteligentes
- Corrección guiada

Semana 4:

- APIs externas
- Procesamiento asíncrono
- Testing y optimizaciones

Entregables:

- Portal web funcionando
- Documentación técnica
- Manual de usuario
- API docs (si aplica)

PREGUNTAS PARA DEFINIR ALCANCE

1. ¿WorldOffice tiene API para importar directamente o es manual desde Excel?
2. ¿Cuántas importaciones masivas hacen los clientes al mes?
3. ¿Necesitan que quede integrado en la plataforma actual o puede ser standalone?
4. ¿Tienen presupuesto definido para desarrollo?
5. ¿Qué tan prioritario es vs otras mejoras?