
















Sistema de Almacenamiento Dual (Local/AWS S3) - EscalaFin MVP

Descripción General

EscalaFin MVP incluye un sistema de almacenamiento de archivos dual que permite elegir entre almacenamiento local en el servidor o almacenamiento remoto en AWS S3. Esta funcionalidad es completamente configurable desde el panel de administración.

Arquitectura del Sistema

Componentes Principales

	Sistema de Almacenamiento	
	Configuración	
	storage-config.ts	# Configuración unificada
	storage-service.ts	# Servicio abstracto
	Almacenamiento Local	
	Directorio: /uploads	
	API: /api/files/serve/[...path]	
	Almacenamiento S3	
	AWS SDK v3	
	URLs firmadas (1 hora)	
	API Unificada	
	/api/files/upload	# Subir archivos
	/api/files/[id]	# Obtener/Eliminar archivo
	/api/files/list	# Listar archivos
	/api/ admin /storage/	# Configuración admin

Base de Datos

```
-- Tabla de archivos con soporte dual
CREATE TABLE "File" (
  "id" TEXT PRIMARY KEY,
  "fileName" TEXT NOT NULL,
  "originalName" TEXT NOT NULL,
  "filePath" TEXT NOT NULL, -- S3 key o path local relativo
  "fileSize" INTEGER NOT NULL,
  "mimeType" TEXT NOT NULL,
  "category" TEXT DEFAULT 'general',
  "description" TEXT,
  "clientId" TEXT,
  "uploadedById" TEXT NOT NULL,
  "storageType" TEXT DEFAULT 'local', -- 'local' o 's3'
  "createdAt" TIMESTAMP DEFAULT now(),
  "updatedAt" TIMESTAMP NOT NULL
);
```

Configuración del Sistema

Variables de Entorno





```
# Configuración de Almacenamiento (Dual: Local/S3)
STORAGE_TYPE=local # 'local' o 's3'

# Configuración Local
LOCAL_UPLOAD_DIR=/home/ubuntu/escalafin_mvp/uploads
LOCAL_BASE_URL=/api/files/serve
LOCAL_MAX_FILE_SIZE=10

# Configuración AWS S3 (solo si STORAGE_TYPE=s3)
AWS_BUCKET_NAME=escalafin-uploads
AWS_REGION=us-east-1
AWS_FOLDER_PREFIX=escalafin-mvp/
S3_MAX_FILE_SIZE=10
AWS_ACCESS_KEY_ID=tu_access_key_aqui
AWS_SECRET_ACCESS_KEY=tu_secret_key_aqui
```

Panel de Configuración Admin

Acceda a `/admin/storage` para:

-  Cambiar entre almacenamiento local y S3
-  Configurar parámetros específicos
-  Probar conectividad
-  Guardar configuraciones

Uso del Sistema

1. Subir Archivos

```
// Componente FileUpload
import { FileUpload } from '@components/files/file-upload'

<FileUpload
  onUploadComplete={(files) => console.log(files)}
  maxFiles={5}
  maxSize={10}
  acceptedTypes={['.pdf', '.jpg', '.jpeg', '.png']}
  clientId="cliente_id"
  category="identification"
  description="Documento de identidad"
/>
```

2. Gestionar Archivos

```
// Componente FileManager
import { FileManager } from '@components/files/file-manager'

<FileManager
  clientId="cliente_id"           // Filtrar por cliente
  category="contracts"           // Filtrar por categoría
  allowUpload={true}             // Permitir subida
  compact={false}                // Vista completa
/>
```

3. API REST

```
// Subir archivo
const formData = new FormData()
formData.append('file', file)
formData.append('category', 'identification')
formData.append('clientId', 'cliente_id')

const response = await fetch('/api/files/upload', {
  method: 'POST',
  body: formData
})

// Listar archivos
const files = await fetch('/api/files/list?category=contracts&clientId=123')

// Obtener archivo
const file = await fetch('/api/files/archivo_id')

// Eliminar archivo
await fetch('/api/files/archivo_id', { method: 'DELETE' })
```



Categorías de Archivos

El sistema incluye categorías predefinidas:

Categoría	Descripción	Ejemplos
identification	Documentos de identidad	INE, Pasaporte, Licencia
income_proof	Comprobantes de ingresos	Nómina, Estados financieros
address_proof	Comprobantes de domicilio	CFE, Agua, Teléfono
contracts	Contratos y acuerdos	Contratos de préstamo
payments	Comprobantes de pago	Tickets, Transferencias
reports	Reportes del sistema	PDFs generados
general	Archivos generales	Otros documentos

Seguridad y Permisos




Control de Acceso

```
// Roles y permisos
ADMIN: // Ve todos los archivos, puede eliminar cualquiera
- Acceso total a todos los archivos
- Configuración del sistema
- Eliminación de archivos

ASESOR: // Ve archivos de clientes asignados
- Archivos subidos por él
- Archivos de clientes asignados
- No puede ver archivos de otros asesores

CLIENTE: // Solo sus propios archivos
- Solo archivos asociados a su cuenta
- No puede ver archivos de otros clientes
```

URLs Firmadas (Solo S3)

-  Expiración: 1 hora
-  Acceso autenticado requerido
-  Logs de auditoría automáticos

Migración Entre Sistemas

De Local a S3

```
# 1. Configurar credenciales AWS
export AWS_ACCESS_KEY_ID="tu_key"
export AWS_SECRET_ACCESS_KEY="tu_secret"

# 2. Crear bucket S3
aws s3 mb s3://escalafin-uploads

# 3. Actualizar .env
STORAGE_TYPE=s3
AWS_BUCKET_NAME=escalafin-uploads

# 4. Migrar archivos existentes (script personalizado)
node scripts/migrate-to-s3.js
```

De S3 a Local

```
# 1. Crear directorio local
mkdir -p /home/ubuntu/escalafin_mvp/uploads





# 2. Actualizar .env
STORAGE_TYPE=local
LOCAL_UPLOAD_DIR=/home/ubuntu/escalafin_mvp/uploads

# 3. Descargar archivos de S3 (script personalizado)
node scripts/migrate-to-local.js
```



Monitoreo y Mantenimiento

Métricas Importantes

-  **Espacio utilizado:** Monitor local/S3
-  **Cantidad de archivos:** Por categoría
-  **Tamaño promedio:** Optimización
-  **Velocidad de carga:** Performance

Logs de Sistema

```
# Ver logs de almacenamiento
tail -f logs/storage.log

# Monitorear subidas
grep "upload" logs/api.log

# Errores de S3
grep "S3 Error" logs/error.log
```

Mantenimiento Regular

```
# Limpiar archivos huérfanos (sin registro DB)
npm run cleanup:orphaned-files

# Verificar integridad de archivos
npm run verify:file-integrity

# Optimizar almacenamiento
npm run optimize:storage
```



Solución de Problemas

Problemas Comunes

1. Error de permisos (Local)

```
# Verificar permisos del directorio
ls -la /home/ubuntu/escalafin_mvp/uploads

# Corregir permisos
chmod 755 /home/ubuntu/escalafin_mvp/uploads
chown -R ubuntu:ubuntu /home/ubuntu/escalafin_mvp/uploads
```

2. Error de credenciales S3

```
# Verificar credenciales
aws configure list

# Probar acceso al bucket
aws s3 ls s3://escalafin-uploads
```

3. Archivos no se muestran

```
# Verificar base de datos
psql $DATABASE_URL -c "SELECT COUNT(*) FROM \"File\";"

# Verificar configuración
curl http://localhost:3000/api/admin/storage/config
```



Componentes de UI

FileUpload

- **Propósito:** Subir archivos con drag & drop
- **Responsive:** Funciona en móvil y desktop
- **Progress:** Barra de progreso en tiempo real
- **Validación:** Tamaño, tipo y cantidad

FileManager

- **Vista completa:** Grid y lista
- **Filtros:** Por categoría, cliente, fecha
- **Descargas:** Directas y seguras
- **Eliminación:** Con confirmación

StorageConfig (Admin)

- **Configuración:** Cambio entre local/S3
- **Pruebas:** Test de conectividad
- **Métricas:** Uso y rendimiento



Funcionalidades Futuras

En Desarrollo

- [] Compresión automática de imágenes
- [] Thumbnails automáticos
- [] Versionado de archivos
- [] Backup automático entre storages
- [] CDN integration
- [] Escaneo antivirus

Roadmap

- [] Azure Blob Storage support
 - [] Google Cloud Storage support
 - [] Encryption at rest
 - [] GDPR compliance tools
 - [] Advanced analytics
-

Soporte Técnico

Para soporte técnico con el sistema de almacenamiento:

1. **Documentación:** Revise esta guía completa
2. **Logs:** Consulte los logs del sistema
3. **Panel Admin:** Use `/admin/storage` para diagnósticos
4. **Testing:** Execute las pruebas de conectividad

Versión del Sistema: 2.1.0

Última actualización: Septiembre 2025

Compatibilidad: Next.js 14, Prisma 6, AWS SDK v3