

Gestión de Riesgos para Activos Digitales

Descripción del Proyecto

Identificación y gestión automatizada de riesgos en activos digitales, que utiliza modelos de lenguaje avanzados ejecutados localmente. La herramienta proporciona:

- Generación automática de perfiles de riesgo
- Análisis de impactos potenciales
- Recomendaciones de mitigación alineadas con ISO 27001
- Interfaz intuitiva para gestión de casos

Tecnologías Implementadas

Frontend

- **React 18** con **Vite** (entorno de desarrollo rápido)
- **Ant Design** (biblioteca de componentes UI profesional)
- **Axios** (para comunicación con el backend)

Backend

- **Flask** (servidor web ligero en Python)
- **Ollama** (infraestructura local para modelos LLM)
- **LangChain** (orquestración de prompts avanzados)
- **LLAMA** (modelo de lenguaje para recomendaciones)

Requisitos del Sistema

Dependencias principales

- Visual Studio Code <https://code.visualstudio.com>
- Node.js v18+ (para el frontend) <https://nodejs.org/en/download>
- Python 3.9+ (para el backend o la versión más reciente) <https://www.python.org/downloads/>
- Ollama instalado localmente <https://ollama.com>
- Git (para clonar el repositorio) <https://git-scm.com/downloads>

Instalación y Configuración

1. Clonar el repositorio

```
git clone https://github.com/OscarJimenezFlores/CursoAuditoria/tree/main/AuditoriaRiesgos
cd AuditoriaRiesgos
```

2. Configuración del FRONTEND

```
cd frontend  
npm install  
npm run dev
```

El FRONT estará disponible en: <http://localhost:5173>

3. Configuración del Backend

Para usuarios macOS

```
brew install ollama  
ollama pull llama2:7b  
ollama run llama2:7b
```

Para usuarios Windows

1. Descargar el instalador desde ollama.com (<https://ollama.com>)
2. O puede ejecutar en la terminal de VsCode / PowerShell:

```
ollama pull llama2:7b  
ollama run llama2:7b
```

Opcionalmente, puede verificar el estado del servicio:

```
tasklist | findstr ollama
```

4. Iniciar el BACKEND servidor Flask

```
python app.py
```

Nota: Si encuentras problemas con la versión de Python, prueba con el número de versión Python que instalaste, por ejemplo:

```
python3.9 app.py
```

Solución de Problemas Comunes

Errores de dependencias

Ejecuta:

```
npm ci --force # Para frontend  
pip install --upgrade -r requirements.txt # Para backend
```

Problemas de compatibilidad

Asegúrate de tener instalado

- Node.js v18+
- Python 3.9+
- Ollama versión estable más reciente

Terminar Procesos en MAC terminal

```
# Para frontend (React)  
pkill -f "npm run dev"  
  
# Para backend (Flask)  
pkill -f "python app.py"  
  
# Para Ollama  
pkill -f "ollama"
```

Terminar Procesos en Windows (VSCode/PowerShell)

```
# Para frontend  
taskkill /F /IM node.exe  
  
# Para backend  
taskkill /F /IM python.exe  
  
# Para Ollama  
taskkill /F /IM ollama.exe
```

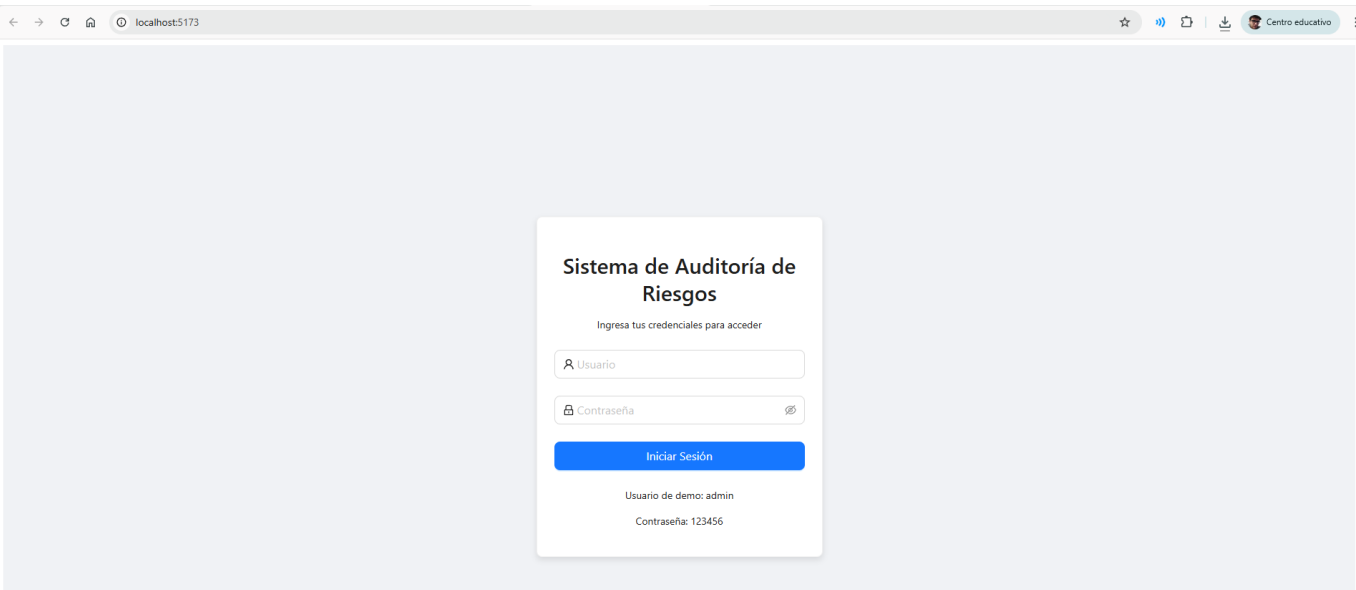
Informe de Auditoría de Sistemas - Examen de la Unidad I

- Nombres y apellidos: Jesus Humberto Escalante Alanoca
- Fecha: 10/09/2025
- URL GitHub: https://github.com/JesusEscalante/ExamenU1_Auditoria.git

1. Proyecto de Auditoría de Riesgos

Login

- Evidencia



- Descripción: Se descargó NodeJS en su versión más reciente luego se ejecuto los siguientes comnados: npm install y npm run dev.

Motor de Inteligencia Artificial

- Evidencia

```
PS C:\Users\HP\Documents\CursoAuditoria-main\AuditoriaRiesgos> ollama pull llama2:7b
pulling manifest
pulling 8934d96d3f08: 100% ▢ 3.8 GB
pulling 8c17c2ebb0ea: 100% ▢ 7.0 KB
pulling 7c23fb36d801: 100% ▢ 4.8 KB
pulling 2e0493f67d0c: 100% ▢ 59 B
pulling fa304d675061: 100% ▢ 91 B
pulling 42ba7f8a01dd: 100% ▢ 557 B
verifying sha256 digest
writing manifest
success
PS C:\Users\HP\Documents\CursoAuditoria-main\AuditoriaRiesgos>
```

```
PS C:\Users\HP\Documents\CursoAuditoria-main\AuditoriaRiesgos> ollama run llama2:7b
>>> /?
Available Commands:
  /set      Set session variables
  /show     Show model information
  /load <model> Load a session or model
  /save <model> Save your current session
  /clear    Clear session context
  /bye      Exit
  /?, /help Help for a command
  /? shortcuts Help for keyboard shortcuts

Use "" to begin a multi-line message.
>>> Send a message (/? for help)
```

- Descripción: descargar e instalar el modelo de lenguaje Llama 2 con 7 mil millones de parámetros usando Ollama.

2. Hallazgos

Activo 1: (Servidor de Base de Datos)

- Evidencia:

Servidor de base de datos	Pérdida de Servidor de base de datos	Pérdida de información valiosa relacionada con Servidor de base de datos	Copias de seguridad periódicas
---------------------------	--------------------------------------	--	--------------------------------

- Condición: Pérdida de Servidor de base de datos
- Recomendación: Copias de seguridad periódicas
- Riesgo: Probabilidad (Media)

Activo 2: (API Transacciones)

- Evidencia:

API Transacciones	Pérdida de API Transacciones	Pérdida de información valiosa relacionada con API Transacciones	Monitoreo continuo de accesos
-------------------	------------------------------	--	-------------------------------

- Condición: Pérdida de API Transacciones
- Recomendación: Monitoreo continuo de accesos
- Riesgo: Probabilidad (Baja)

Activo 3: (Aplicación Web de Banca)

- Evidencia:

Aplicación Web de Banca	Pérdida de Aplicación Web de Banca	Pérdida de información valiosa relacionada con Aplicación Web de Banca	Cifrado de datos sensibles
-------------------------	------------------------------------	--	----------------------------

- Condición: Pérdida de Aplicación Web de Banca
- Recomendación: Cifrado de datos sensibles
- Riesgo: Probabilidad (Media)

Activo 4: (Servidor de Correo)

- Evidencia:

Servidor de Correo	Pérdida de Servidor de Correo	Pérdida de información valiosa relacionada con Servidor de Correo	Monitoreo continuo de accesos
--------------------	-------------------------------	---	-------------------------------

- Condición: Pérdida de Servidor de Correo
- Recomendación: Monitoreo continuo de accesos
- Riesgo: Probabilidad (Baja)

Activo 5: (Firewall Perimetral)

- Evidencia:

Firewall Perimetral	Pérdida de Firewall Perimetral	Pérdida de información valiosa relacionada con Firewall Perimetral	Implementación de controles de acceso físico
---------------------	--------------------------------	--	--

- Condición: Pérdida de Firewall Perimetral
- Recomendación: Implementación de controles de acceso físico

- Riesgo: Probabilidad (Media)