

IA local para PyMEs · Módulo 11

Multiusuario y operación

Tres empleados, tres permisos, un sistema. Trazas atribuidas como producto y Git que separa plantilla de cliente.

Qué se construye en este módulo

Identidad federada, grupos, cuotas y aislamiento

Dar de alta tres usuarios con permisos distintos

Trazas atribuidas como producto vendible

Registrar quién, qué, con qué fuente y a qué costo

Git para el servicio: plantilla, derivados y secretos fuera

Separar repo plantilla de repos por cliente

11.1 · Identidad federada, grupos, cuotas y aislamiento

Identidad federada, grupos, cuotas y aislamiento

El gerente ve todo. La contadora ve contabilidad y contratos, y **aprueba** registros. El vendedor ve inventario y precios, y nada más. Esa frase, convertida en configuración, es este módulo.

En una tabla

Capa	Dónde se define	Qué garantiza
Identidad	proveedor SSO (3.2): usuario, MFA, grupos	quién es
Autorización por grupos	el proxy pasa los grupos; cada servicio los lee de la cabecera	qué puede ver
Permisos en el índice	permisos && grupos antes de recuperar (6.3)	qué documentos entran a texto
Aprobaciones	la cola (10.4) filtra por grupo	qué puede autorizar
Cuotas	por usuario: peticiones/minuto, tokens/día	que uno no sature a los de
Aislamiento de datos	conversaciones, borradores y trazas con <code>usuario_id</code> ; nadie lee las de otro	privacidad entre empleados

El comando, copiable

```
# cuota.py - límite por usuario, en el nodo de servicios, antes de llamar al modelo
LIMITES = {"gerencia": {"rpm": 30, "tokens_dia": 400_000}, "contabilidad": {"rpm":
    20, "tokens_dia": 250_000},
    "ventas": {"rpm": 10, "tokens_dia": 100_000}}
def permitido(usuario, grupos, tokens_estimados, db):
    lim = max((LIMITES[g] for g in grupos if g in LIMITES), key=lambda l: l["rpm"],
        default=LIMITES["ventas"])
    if db.peticiones_ultimo_minuto(usuario) >= lim["rpm"]: return False, "límite por
        minuto"
    if db.tokens_hoy(usuario) + tokens_estimados > lim["tokens_dia"]: return False, "
        cuota diaria"
    return True, ""
```

Si no corre desde cero con un comando, no entra al curso.

Cuándo NO usar esto

Identidad federada, grupos, cuotas y aislamiento

- No implemente permisos «en el prompt» («no muestres nómina al vendedor»): el modelo no es un control de acceso.
- No comparta un usuario entre empleados «para simplificar»: se pierde la atribución y la ley (2.3).

Qué se degrada en cada perfil

A · sin GPU

Un usuario a la vez:
las cuotas sirven para
turnar, no para repar-
tir

B · 8–16 GB VRAM

Cuotas ajustadas a
3–5 usuarios

C · 24 GB+

Holgura

11.2 · Trazas atribuidas como producto vendible

Trazas atribuidas como producto vendible

Un cliente no puede ver «la IA». Puede ver un registro que diga: *el 14 de agosto a las 10:32, la contadora consultó la garantía del contrato 2026-014; el sistema recuperó la cláusula 7 (página 3); respondió citándola; costó 3 400 tokens y 4,1 segundos.* Eso se puede auditar, facturar y mejorar. Sin eso, el sistema es una caja negra que un día alguien apaga.

En una tabla

Para quién	Reporte	Frecuencia
Gerente	uso por persona, tareas completadas, ahorro estimado de horas	mensual
Contadora	qué se registró automático y qué aprobó ella	semanal
Implementador	tasas de fallo por estado, latencias, cuota consumida	diaria (canario)
Auditoría	quién vio qué documento	a demanda

El comando, copiable

```
{
  "id": "...tr_01J",
  "cuando": "2026-08-14T10:32:07-05:00",
  "usuario": "contadora",
  "grupos": [
    "contabilidad"
  ],
  "tarea": "consulta",
  "estado_final": "respondida",
  "pasos": [
    {
      "n": 1,
      "rol": "consultor",
      "modelo": "<modelo>@<version>",
      "tokens_in": 3100,
      "tokens_out": 300,
      "ms": 4100,
      "fuentes": [
        "contrato-2026-014#7"
      ],
      "herramientas": [
        "buscar_documentos"
      ]
    }
  ],
  "costo": {
    "tokens": 3400,
    "segundos": 4.1
  },
  "aprobaciones": [],
  "errores": []
}
```

Si no corre desde cero con un comando, no entra al curso.

Cuándo NO usar esto

Trazas atribuidas como producto vendible

- No guarde el contenido de los documentos en la traza «por si acaso».
- No mida «satisfacción» con un pulgar: mida tareas completadas sin corrección humana.

11.3 · Git para el servicio: plantilla, derivados y secretos fuera

Git para el servicio: plantilla, derivados y secretos fuera

Todo lo que se construyó es texto (0.2). Se organiza en dos niveles:

En una tabla

Se versiona	No se versiona
código, esquemas, contratos, máquinas de estado docker-compose, configuración de proxy y SSO sin claves documentación, runbooks, políticas set de prueba anotado (en el repo privado del cliente) exportaciones de flujos (9.1)	.env y cualquier secreto pesos de modelos índice vectorial, bases de dato trazas documentos del cliente

El comando, copiable

```
# .env.example – versionado, sin valores
POSTGRES_PASSWORD=
SSO_CLIENT_SECRET=
RESTIC_PASSWORD_FILE=/run/secrets/restic
# .gitignore
.env
*.key
config/secrets/
```

Si no corre desde cero con un comando, no entra al curso.

Cuándo NO usar esto

Git para el servicio: plantilla, derivados y secretos fuera

- No haga un repo por cliente copiando y pegando: en seis meses son seis productos distintos.
- No versione el set de prueba en el repo de la plantilla: son datos del cliente.

Entregables del módulo

- `labs/11-01/prueba-usuarios.md` con las 12 consultas y su resultado.
- `traza.py`, `reporte.py`, y un reporte con datos del laboratorio.
- los dos repos, el gancho, y `labs/11-03/propagacion.md`.

**Tres empleados, tres
permisos, un sistema.**