

INSTRUCTIVO SISTEMA AUTOMATIZADO DE GESTIÓN DE COMPRAS, FACTURAS, ENVIOS Y USUARIOS

Integrantes:

Josué Vásquez Latin

Sofía Alvarado

Criss Valdes

Fecha: 24 de julio de 2025

Información General

Repositorio GitHub:

<https://github.com/Josue19688/IRSI-SISAP>

Autor principal: Josué Vásquez Latin

Fecha: 24 de julio de 2025

Lenguajes utilizados: Python, Bash, PowerShell

Automatización: cron en Linux y Task Scheduler en Windows

Requisitos de entorno: Python 3.8+, Git Bash, LaTeX, PowerShell 5.1+

1. Instalación y Configuración del Entorno

Clonar el Repositorio

git clone <https://github.com/Josue19688/IRSI-SISAP>.git

cd IRSI-SISAP

Estructura principal del proyecto

IRSI-SISAP/

```
├── core/
│   ├── compras/
│   ├── facturacion/
│   ├── envios/
│   └── usuarios/
├── infrastructure/
│   ├── cron/
│   ├── task_scheduler/
│   └── reportes/
├── data/
│   ├── csv/
│   ├── facturas/
│   └── logs/
├── docs/
├── main.py
└── README.md
```

Requisitos por lenguaje

Python

pip install faker

PowerShell (Windows)

- Tener PowerShell 5.1 o superior
- Ejecutar scripts permitiendo la política de ejecución:

Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass

Bash

- Instalar Git Bash en Windows (recomendado)
- Verificar que bash esté en el PATH

LaTeX

- Instalar **MiKTeX** en Windows o **TeX Live** en Linux
- Verificar compilación:

pdflatex --version

2. Uso del Script Principal (main.py)

El script main.py permite ejecutar de forma unificada todas las tareas mediante **banderas** (flags). Cada bandera representa una acción concreta del sistema.

Ejecución básica

python main.py -g "Generar CSV compras"

Banderas disponibles

Bandera	Acción	Descripción
-g o --generar	Generar compras	Crea archivos CSV con datos ficticios de compras
-f o --factura	Generar facturas	Crea facturas PDF desde los CSV generados
-e o --enviar	Enviar facturas	Envía los PDFs por correo según archivo pendientes_envio.csv
-i o --informe	Generar informe	Crea un resumen automático de los logs generados
-u o --usuarios	Crear usuarios	Crea usuarios locales desde empleados.csv con contraseñas seguras

Ejemplo de ejecución secuencial

python main.py -g "Generar CSV compras"

python main.py -f "Generar Facturas"

python main.py -e "Enviar PDF"

python main.py -i "Resumen"

python main.py -u

3. Automatización con Cron o Task Scheduler

Windows (Task Scheduler)

1. Abre **Programador de tareas** → Crear tarea básica.
2. Establece nombre: IRSI_SISAP_Tarea_Diaria
3. Programa ejecución diaria:
 - 1:00 AM → Ejecutar facturas
 - 2:00 AM → Enviar correos
 - 2:30 AM → Generar resumen

4. Comando a ejecutar:
"C:\ruta\al\python.exe" "C:\ruta\al\proyecto\main.py" -f "Facturas"
5. Repetir con las banderas -e y -i para tareas secuenciales.

Linux (cron)

Editar cron con:

crontab -e

Agregar:

0 1 * * * /usr/bin/python3 /ruta/main.py -f "Facturas"

0 2 * * * /usr/bin/python3 /ruta/main.py -e "Envio"

30 2 * * * /usr/bin/python3 /ruta/main.py -i "Informe"

4. Formato del Archivo empleados.csv (para PowerShell)

Debe colocarse en data/empleados.csv con esta estructura:

Nombre,Apellido,Correo

Luis,Martínez,luis@gmail.com

Ana,López,ana@hotmail.com

Cada fila generará un usuario con contraseña segura y log personalizado.

5. Directorios Clave

Ruta	Descripción
data/csv/	Archivos de compras generados y pendientes_envio.csv
data/facturas/	PDFs generados por LaTeX
data/logs/	Logs diarios, de envíos y usuarios
core/	Scripts por dominio funcional (compras, envíos, usuarios)
infraestructure/	Automatización, tareas programadas, reportes
docs/	Reporte técnico, capturas, manuales

6. Logs y Reportes

log_diario.log

Contiene la salida de la compilación LaTeX (errores o éxito).

log_envios.log

Registra fecha, archivo enviado, correo destino y estado.

Logs individuales por usuario

Cada nuevo usuario creado tiene su propio log en data/logs/usuario.log con:

- Nombre de usuario
- Correo
- Contraseña
- Fecha de creación

7. Evidencias y Capturas

- Capturas de pantalla están disponibles en:

docs/capturas/



- Ver resultados en PDF: data/facturas/*.pdf
- Logs individuales y resumen: data/logs/

8. Recomendaciones Finales

- Verifica que las rutas de los scripts estén correctas en tu sistema.
- Asegúrate de tener permisos de administrador para crear usuarios.
- No compartas los logs de usuarios sin antes eliminar las contraseñas.
- Usa GitHub como respaldo y versión controlada de tu proyecto.

Contacto

Si tienes dudas o deseas colaborar:

-  advinjosuev899@gmail.com
-  GitHub: [Josue19688](#)