LUIS ALVARO PIMENTEL TACURI

Desarrollador Python Junior | Aplicaciones web, web scraping y automatización de datos

Potosí, Bolivia | luiscontactojob2007@gmail.com

GitHub: github.com/Gokublackssj

PERFIL PROFESIONAL

Estudiante de primer año de Ingeniería en Informática. Desarrollo aplicaciones web con FastAPI, PostgreSQL y Next.js que validan los datos antes de guardarlos, y scrapers con Playwright que exportan a Excel y CSV.

HABILIDADES TÉCNICAS

Enfoque: Validación de datos · Web scraping · Exportación a Excel/CSV · APIs REST · Pruebas automatizadas

Herramientas principales: Python, Playwright, openpyxl, pytest

También usadas en proyectos: FastAPI, Pydantic, SQLAlchemy, PostgreSQL, Next.js, React, TypeScript, Alembic, Docker Compose, Git, GitHub Actions, Ruff

PROYECTOS DESTACADOS

QuoteFlow — Aplicación web de importación y revisión de presupuestos ***Proyecto personal, se ejecuta en local***

Repositorio: github.com/Gokublackssj/quoteflow

- Desarrollé con FastAPI y PostgreSQL una aplicación que importa presupuestos de CSV, Excel y PDF, analiza cada archivo en un proceso aislado con límites de tamaño y tiempo, y marca filas inválidas y duplicadas.
- Diseñé en Next.js y TypeScript un paso de revisión en el que se aceptan, excluyen o corrigen filas; nada se guarda hasta que el usuario confirma.
- Implementé sesiones con cookies y CSRF, contraseñas con Argon2, límite de intentos de acceso y auditoría con correos seudonimizados; 512 pruebas de backend (pytest) y 46 de frontend.
- Añadí un endpoint opcional para n8n, desactivado por defecto; el flujo de ejemplo sigue inactivo.

LeadFlow — Formulario de captación y CRM de leads ***Proyecto personal, se ejecuta en local***

Repositorio: github.com/Gokublackssj/LeadFlow

- Desarrollé una aplicación web que capta leads desde un formulario, los guarda en PostgreSQL y registra el estado, las notas de seguimiento y el historial de cada lead en un CRM.
- Construí una API REST con FastAPI, Pydantic y SQLAlchemy que ignora envíos repetidos (identificadores de evento idempotentes) y correos duplicados por fuente, con búsqueda paginada y filtros por estado.
- Desarrollé el frontend con Next.js y TypeScript; probé el backend con pytest (148 pruebas) y la interfaz con pruebas de extremo a extremo en Playwright (17).

Book Data Extractor Pro — Scraper web en Python ***Proyecto personal***

Repositorio: github.com/Gokublackssj/book-data-extractor-pro

- Desarrollé un scraper con Playwright para un catálogo de práctica paginado (books.toscrape.com) que exporta título, precio, disponibilidad y URL a Excel (openpyxl) y CSV; 20 libros por defecto, ajustable con --limit.
- Convertí los precios con Decimal, eliminé duplicados por URL y marqué los campos faltantes en lugar de descartar filas; añadí pruebas unitarias con pytest sin conexión que se ejecutan en GitHub Actions.

FORMACIÓN ACADÉMICA

Universidad Autónoma Tomás Frías (UATF) — Potosí, Bolivia ***2026 – Actualidad***

Ingeniería en Informática · Estudiante de primer año

IDIOMAS

Español: Nativo | **Inglés:** Básico