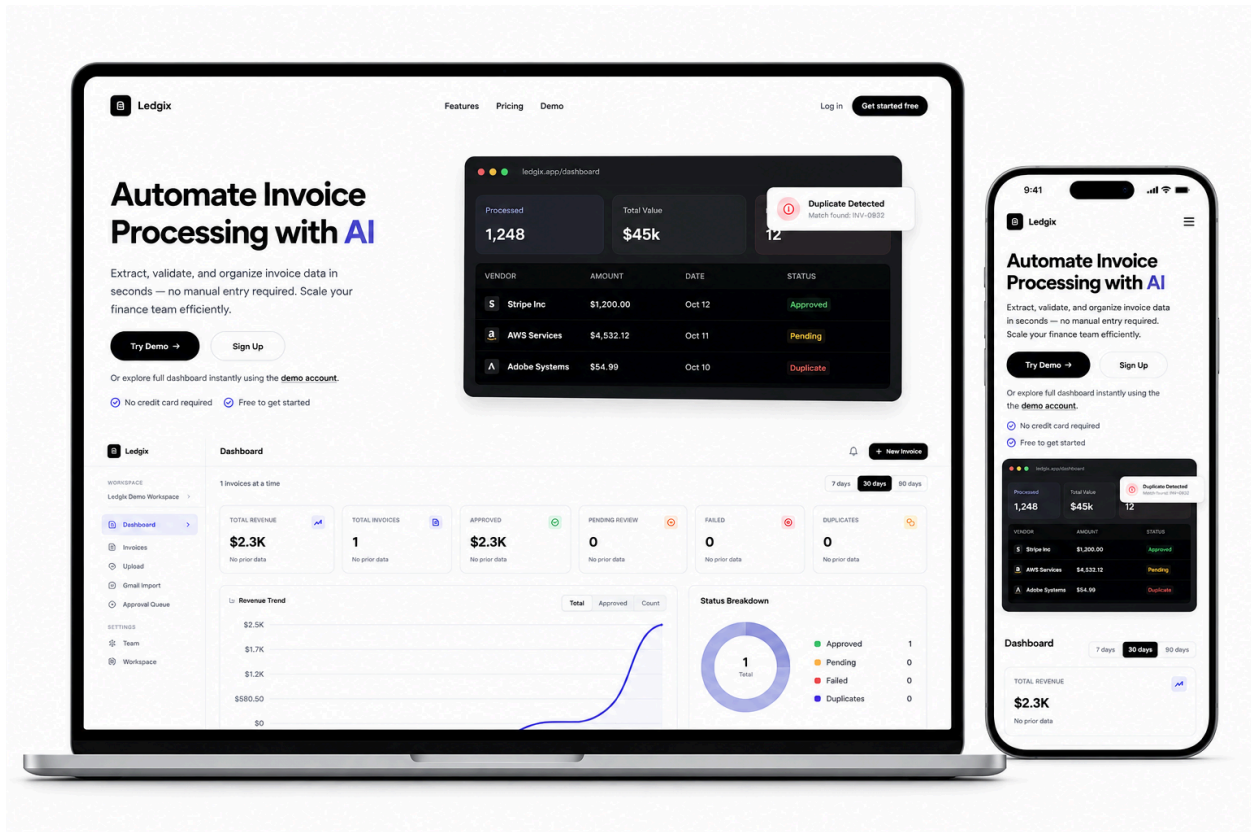


Ledgix AI

An AI-powered invoice automation platform that automatically captures, extracts, validates, reviews, and exports invoice data, eliminating manual processing for finance teams.



Tech Stack

Python • Django • DRF • PostgreSQL • Celery • Redis • Next.js • Typescript • OpenAI • Mindee • Stripe • Docker • nginx • AWS

The Challenge

Finance teams were spending hours manually downloading invoice attachments, entering data into accounting systems, and reviewing invoices for errors. As invoice volumes increased, the process became slow, repetitive, and prone to human error.

Traditional OCR tools extracted text but lacked the intelligence to validate data, understand different invoice layouts, or detect duplicate invoices. Teams still had to manually verify and correct extracted information before processing.

An end-to-end solution was needed that could automatically capture invoices from email, extract structured data with AI, validate results, prevent duplicate payments, and streamline the review and approval process.

Target Users

- Finance teams
- Accounts Payable (AP) departments
- Accounting and bookkeeping firms

Solution

Ledgix AI automated the entire invoice processing workflow, from invoice capture to accounting-ready exports. The platform automatically detects and imports invoices from Gmail or manual uploads, extracts structured data using AI, validates key information, detects duplicate invoices, and routes records through a review and approval workflow. By eliminating repetitive manual tasks, it enables finance teams to process invoices faster, more accurately, and at scale.

Key Features

AI Invoice Extraction: Automatically extracts structured invoice data from PDFs and images.

Gmail Auto Import: Detects and imports invoice attachments directly from Gmail.

Duplicate Detection: Prevents duplicate invoice processing using intelligent matching.

Validation Engine: Verifies extracted data against business rules before approval.

Approval Workflow: Supports collaborative review and approval for finance teams.

Side-by-Side Review: Compare the original invoice with AI-extracted data for quick verification.

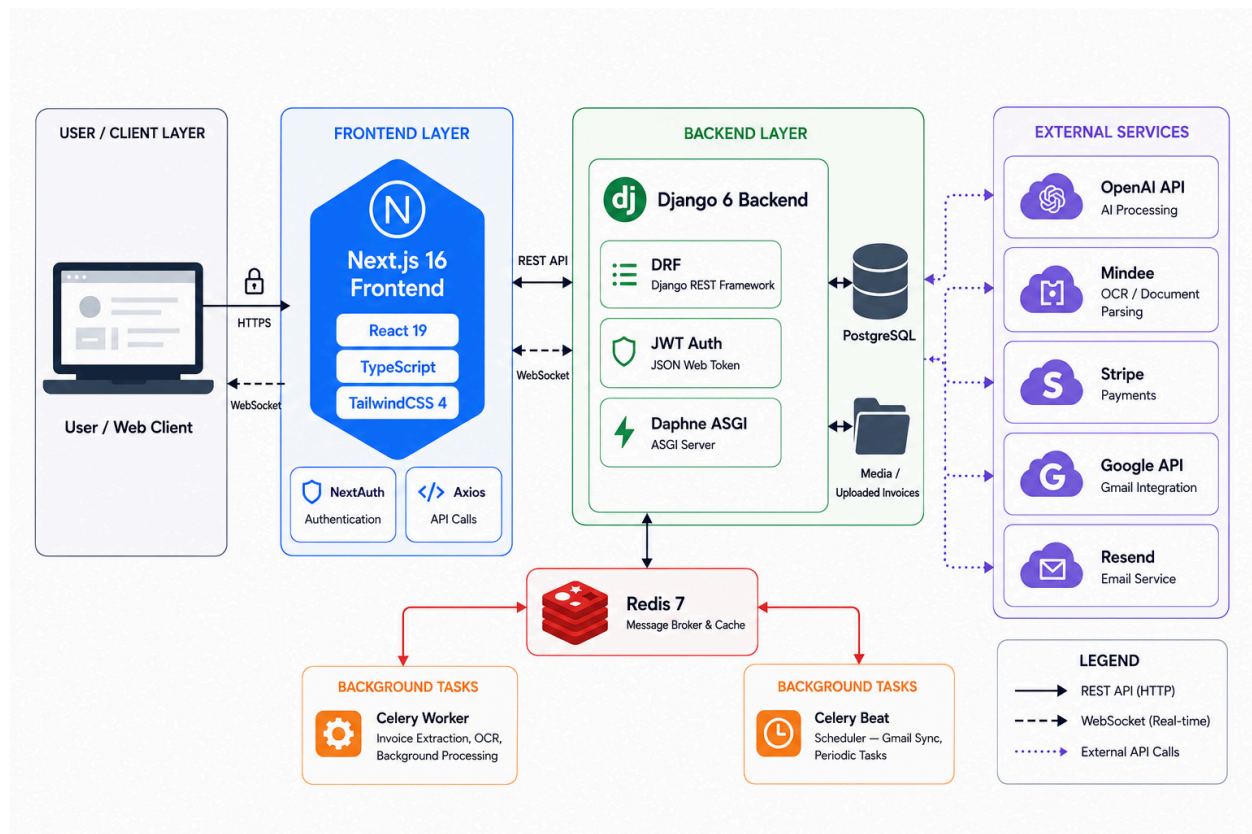
Accounting Exports: Export validated invoices to QuickBooks, Xero, or CSV.

Real-Time Dashboard: Track invoice processing, approvals, and system activity from one place.

Workflow Diagram



Technical Architecture



Challenges & Solutions

Challenge 1: Manual Invoice Processing

Challenge: Finance teams spent significant time manually downloading, uploading, and entering invoice data.

Solution: Automated invoice capture from Gmail and AI-powered data extraction to eliminate repetitive manual work.

Challenge 2: Inconsistent Invoice Formats

Challenge: Vendors used different invoice layouts, making template-based extraction unreliable.

Solution: Integrated Mindee's AI-powered invoice parsing to accurately extract structured data across diverse invoice formats.

Challenge 3: Duplicate Invoice Payments

Challenge: Duplicate invoices were difficult to identify before approval, increasing payment risk.

Solution: Implemented a duplicate detection engine that compares key invoice attributes before processing.

Challenge 4: Maintaining Data Accuracy

Challenge: AI-extracted data required a reliable way to verify accuracy before approval.

Solution: Added validation rules, confidence scores, and a side-by-side review interface for quick verification.

These four challenges are concise, realistic, and align well with the functionality demonstrated by Ledgix AI.

Production & Scalability

Designed using production-grade engineering practices, with a modular architecture that supports reliable deployments, asynchronous processing, and future horizontal scaling. The platform is containerized with Docker, allowing consistent environments across development and production.

Containerized Architecture

The application is deployed as independent Docker services, enabling each component to scale and operate independently.

- **Next.js 16** frontend
- **Django 6** backend
- **Redis** for caching and message brokering
- **Celery Worker** for background processing
- **Celery Beat** for scheduled tasks

Authentication & Authorization

The platform uses **JWT-based authentication** with secure token management through NextAuth and supports **role-based access control (RBAC)** to manage organization-level permissions for Owners, Admins, Members, and Viewers.

Background Processing

Long-running operations are processed asynchronously using **Celery** with **Redis**, ensuring the application remains responsive.

Background jobs include:

- AI invoice extraction via Mindee
- Duplicate invoice detection
- Gmail synchronization
- Scheduled maintenance and cleanup tasks

Real-Time Updates

Invoice processing progress is delivered in real time using **WebSockets (Django Channels + Daphne)**, allowing users to receive status updates without refreshing the application.

Security

The application incorporates several production-ready security practices, including:

- JWT authentication
- CORS protection
- HTTPS support
- Environment-based secrets management
- Organization-level data isolation
- ORM-based SQL injection protection

Database & Performance

Ledgix AI is designed for efficient data processing and future growth using:

- PostgreSQL for persistent data storage
- Redis for caching and background task coordination
- Optimized ORM queries and incremental synchronization to reduce unnecessary database and API operations

Scalability

The architecture is designed to scale horizontally as demand grows.

Component	Scaling Strategy
Frontend	Stateless deployment behind a load balancer
Backend	Multiple ASGI application instances
Celery Workers	Additional workers for higher processing throughput
Redis	Managed Redis service for distributed workloads
PostgreSQL	Managed database with backup and replication support

Deployment

A CI/CD pipeline automates the build, testing, and deployment workflow, enabling reliable and repeatable releases with minimal manual intervention. The architecture is cloud-ready and can be deployed to managed infrastructure, including PostgreSQL, Redis, and container orchestration platforms as scalability requirements grow.

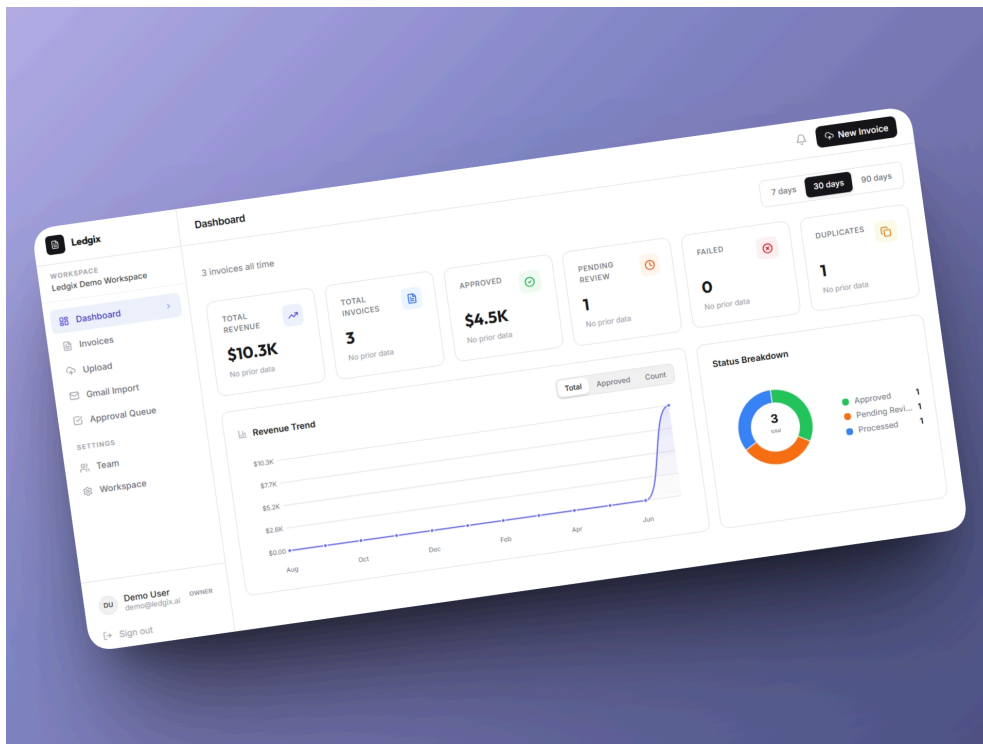
My Responsibilities

As the sole developer, I did the complete design, development, and delivery of the platform, from planning the architecture to deployment. My responsibilities included:

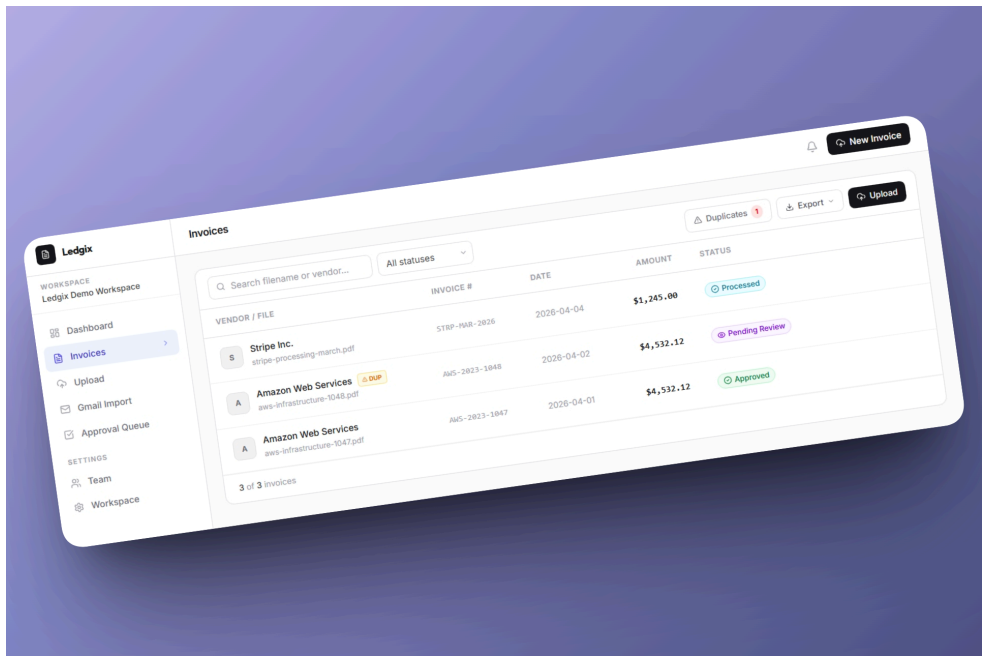
- Designed the overall system architecture and database schema.
- Developed the backend APIs and business logic using Python.
- Integrated Mindee AI for invoice extraction and the Gmail API for automated invoice ingestion.
- Implemented validation, duplicate detection, confidence scoring, and approval workflows.
- Built secure authentication and third-party integrations.
- Optimized database queries, error handling, and application performance.
- Managed deployment, environment configuration, testing, and production readiness.

Gallery

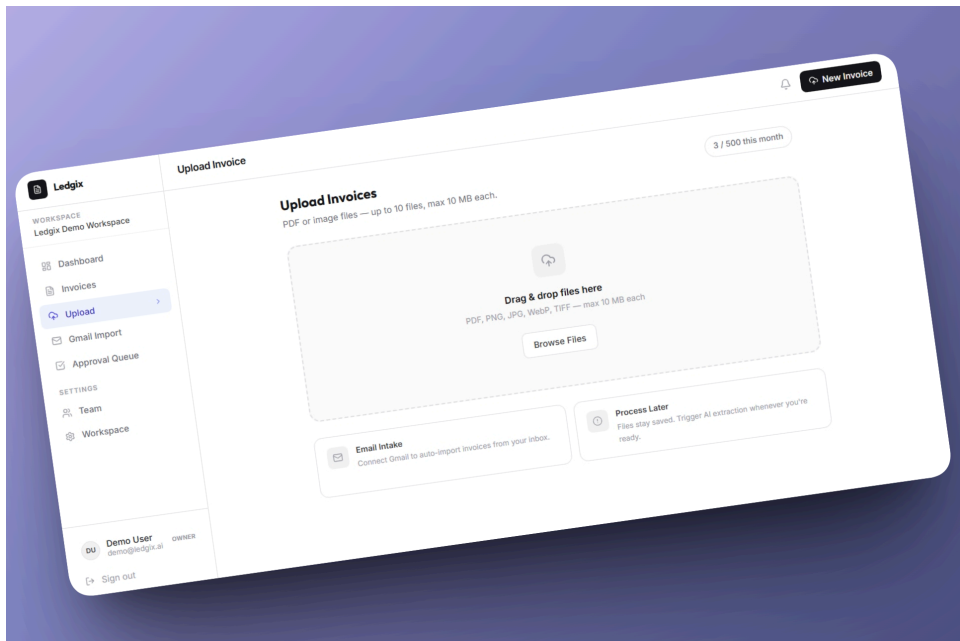
Dashboard



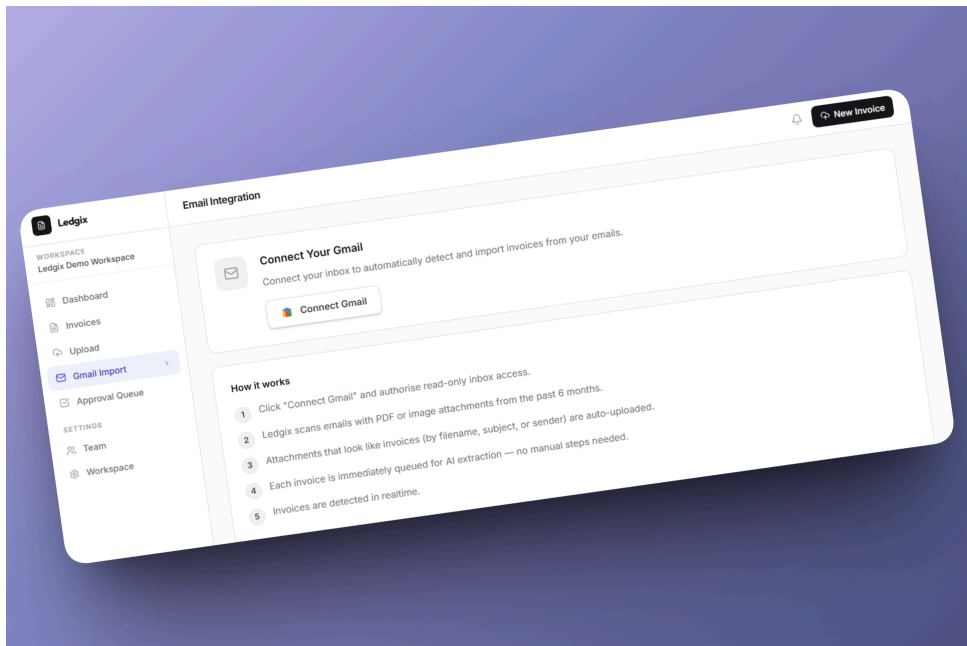
Invoices



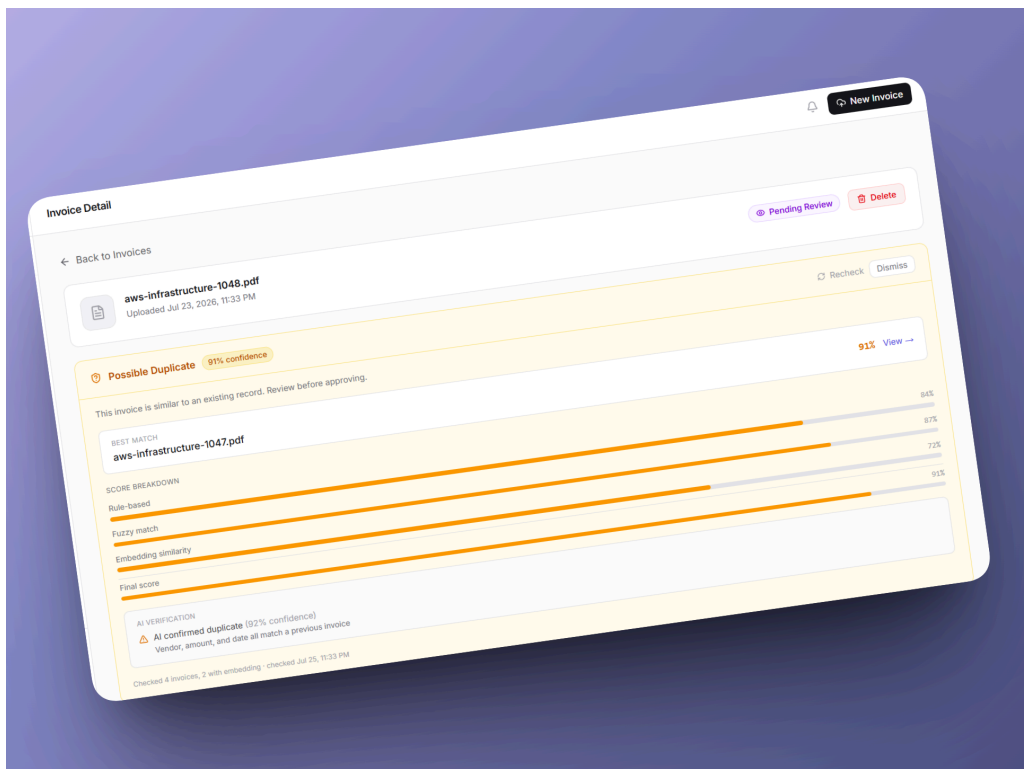
Upload Invoices Manually



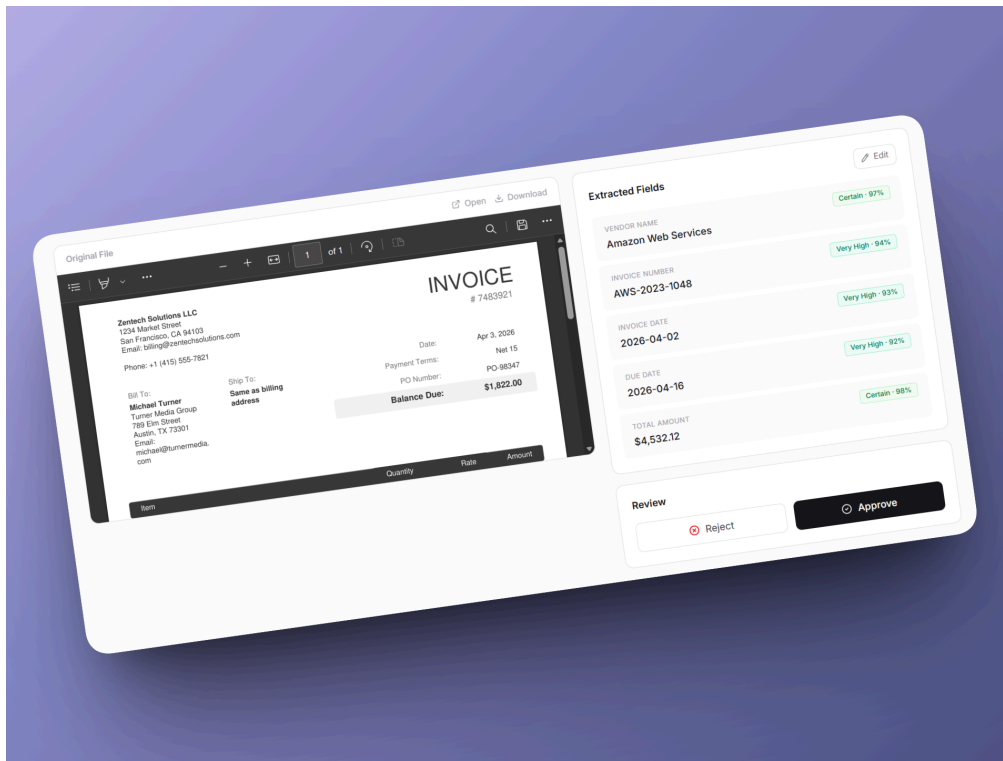
Gmail Integration



Duplicate Invoice Detection



Invoice Preview & Extracted Fields Confidence



Project Links

Live demo:

ledgix.wajahatlabs.com

Github repos:

Frontend: [wajahat-py/ledgix_ai_frontend](https://github.com/wajahat-py/ledgix_ai_frontend)

Backend: [wajahat-py/ledgix_ai_backend](https://github.com/wajahat-py/ledgix_ai_backend)

Final Takeaway

Ledge AI demonstrates how AI can be integrated into business workflows to automate repetitive financial operations without sacrificing accuracy or control. The platform brings together invoice capture, intelligent data extraction, validation, review, and accounting integrations into a unified, production-ready solution designed for real-world use.