

Tech Stack

The **Financial Document Fraud Analyzer** leverages a diverse set of modern technologies to deliver robust, scalable, and explainable fraud detection in financial documents.

Programming Language

- **Python 3.x** — Core language for all modules and scripts.

Optical Character Recognition (OCR)

- **EasyOCR** — Primary OCR engine for extracting text from financial documents.
- **Tesseract / Google Vision OCR** — Supported alternatives configurable for improved OCR accuracy.

Machine Learning & Fraud Detection

- **IsolationForest** — Unsupervised anomaly detection for spotting outlier documents.
- **Custom Rule-Based Heuristics** — Domain-specific rules complement ML detection.
- **XGBoost** (optional) — Supported for advanced model training and fraud prediction.

Explainability & Interpretability

- **SHAP** — Local and global model explanation via Shapley values.
- **LIME** — Model-agnostic interpretability to explain individual predictions.
- **GPT-4** — Generates natural language summaries explaining fraud detection results.

Web Application & User Interface

- **Streamlit** — Interactive and user-friendly dashboard for document upload, analysis, and visualization.

Notifications & Alerting

- **Slack Webhooks** — Real-time fraud alerts pushed to Slack channels for immediate action.

Data Formats & Utilities

- **Regex & NLP** — Extract key financial fields from OCR text using pattern matching and natural language processing.
- **JSON / CSV** — Standard formats for input/output data and reports.

Development & Testing

- **pytest** — Automated unit and integration testing framework.
- **PEP8 Compliance** — Coding standards with type annotations and Google-style docstrings for maintainability.

Containerization & Deployment

- **Docker** — Containerized environment for easy deployment and reproducibility.