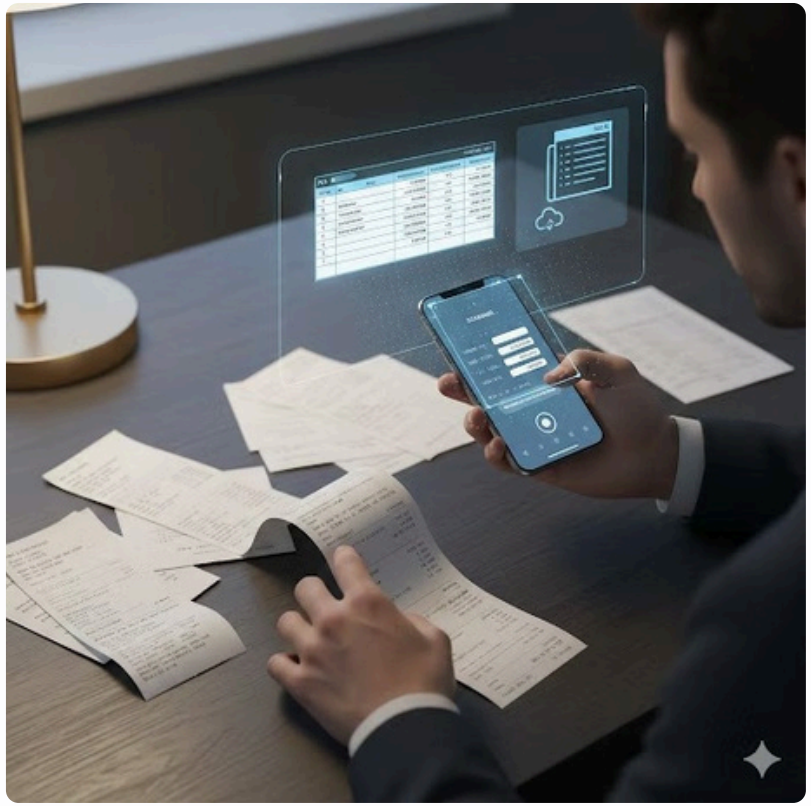
Receipt & Invoice Digitizer

AnAI-PoweredAutomatedExpense&DocumentManagement System that streamlines financial record-keeping.



Mentor
Ms. Santhiya Krishnasamy

Presented By
Tisha Gurjar

Problem Statement

Manual handling of financial documents is inefficient and problematic.

Manual Handling Challenges

- Time-consuming
- Error-prone
- Loss of financial records
- Difficult expense tracking
- Repetitive data entry

Project Overview

This project builds a system that automatically scans, extracts, and digitizes information from receipts and invoices using OCR (Optical Character Recognition) and NLP-based field extraction. The digitized data is stored in a structured format, making it easy to search, analyze, and integrate with accounting or ERP systems.

1. Automated OCR-based text extraction from scanned receipts and invoices.
2. Field-level parsing for date, vendor, amount, tax, and line items.
3. Web-based dashboard for uploading files, reviewing extracted data, and downloading CSV/Excel reports.
4. Error handling & validation (e.g., duplicate receipts, invalid totals).

Implementation Roadmap

Our project delivery will follow a structured, phased approach over 8 weeks, ensuring each module is robustly developed and integrated.

Milestone 1: Weeks 1-2

Document Ingestion & OCR

- Implement file upload.
- Preprocess images for OCR.
- Extract raw text from sample receipts.

Milestone 2: Weeks 3-4

Field Extraction & Validation

- Apply regex + NLP to parse key fields.
- Validate totals and detect duplicates.
- Store structured results in DB.

Milestone 3: Weeks 5-6

Dashboard & Reporting

- Build Streamlit dashboard for upload/review.
- Add CSV/Excel export.
- Display simple analytics (monthly totals).

Milestone 4: Weeks 7-8

Polishing & Integration

- Improve extraction accuracy with template-based parsing.
- Add search/filter in dashboard.
- Optimize DB queries and reports.

Module 1: Core Functionality Objectives

Our initial development phase focuses on establishing the fundamental capabilities required for efficient document digitization and data extraction.

File Upload & Ingestion

Enabling users to easily upload receipt and invoice images or PDFs for automated processing within the system.

Image Preprocessing for OCR

Implementing algorithms to enhance the quality of uploaded documents, maximizing the accuracy of subsequent text recognition.

Optical Character Recognition (OCR)

Developing robust OCR capabilities to precisely extract raw textual data from all scanned receipts and invoices for further analysis.

Core Functionality Workflow

This detailed workflow illustrates the essential steps from initial document ingestion to final management, highlighting the automated processes and AI integration at each stage.

User Interface & Application Control Module

This module acts as the central entry point and controller, ensuring a seamless and intuitive user experience by managing application flow and user interactions.

Application Layout

Configures the **Streamlit** application layout for a clean, organized, and responsive user interface

API Key & Access

Handles Gemini API key(**Gemini 2.5 Flash**) input, validation, and restricts access until a valid key is provided.

Tab Organization

Organizes the application into **Upload, Dashboard, and History** tabs for clear workflow separation.

Receipt Display

Displays stored receipts and detailed line items through **interactive tables**.

Module Coordination

Controls data flow and interaction between **OCR, AI parsing, and database** modules

Core Features

Uses **session state** for API verification, persistent sidebar settings, and interactive selections.

This module is paramount in controlling the overall flow and user experience of the Digitizer application.

Document Ingestion & OCR Processing Module

This module serves as the initial gateway for all documents, transforming raw inputs into structured, machine-readable text.

Document Upload & Conversion

Supports multiple file formats (JPG, JPEG, PNG, PDF) and converts multi-page PDFs into individual images for processing..

Image Preprocessing

Applies critical enhancements like resizing, denoising, and thresholding to optimize image quality for OCR accuracy.

Tesseract OCR Extraction

Extracts raw textual content using Tesseract OCR, optimized for financial documents such as receipts and invoices.

Text Cleaning & Normalization

Cleans and standardizes OCR output to remove noise and prepare text for AI-based parsing.

Ultimately, this foundational module converts physical or digital documents into usable, structured text, ready for intelligent data extraction.

AI-Based Data Extraction & Validation Module

This module is the intelligence layer of the project, leveraging advanced AI to transform unstructured OCR text into structured, reliable financial data for seamless integration and analysis.

AI-Powered Data Extraction

- Sends cleaned OCR text to Gemini AI using strict JSON prompting for structured output.

Key Field Identification

- Extracts essential fields including vendor name, date, total amount, tax, and line items.

Rigorous Data Validation

- Verifies extracted data through numeric checks, mandatory field validation, and line-item consistency.

Quality Filtering

- Discards incomplete or low-confidence extractions to maintain data reliability.

Deterministic AI Output

- Ensures consistent results using deterministic AI settings (temperature = 0) with JSON recovery logic.

Field Normalization

- Standardizes extracted values into consistent formats for accurate storage and processing.

Data Storage, Duplicate Detection & Management Module

This module ensures data integrity, persistence, and efficient retrieval, forming the backbone of the Digitizer's reliable record-keeping.

Database Management

- Initializes and manages the **SQLite database** for robust operations.
- Securely stores both **receipt-level and line-item data**.
- Efficiently retrieves stored receipts and items for display across the application.

Data Integrity & Quality

- Performs **early duplicate detection** using OCR text to prevent redundant processing.
- Maintains **strong relational integrity** between receipts and their individual line items.

Robust Storage Features

- Utilizes a **normalized database schema** for optimal structure and efficiency.
- Ensures **secure and persistent local storage** of all financial documents.

Ultimately, this module guarantees reliable and duplicate-free data storage, critical for accurate financial record management.

Milestone 2: Field Extraction & Validation

In this milestone, we implemented a hybrid NLP and regex-based extraction pipeline with validation logic to ensure accuracy, detect duplicates, and store structured receipt data in a persistent database.

Field Extraction using Regex + NLP

- OCR text is processed using **NLP (LLM-based parsing)** to understand context and extract fields such as vendor name, date, line items, tax, and total.
- **Regex patterns** are used alongside NLP to accurately identify numeric values like dates, prices, quantities, and totals across different receipt formats.

Validate Totals and Detect Duplicates

- Extracted financial values are **validated by cross-checking subtotal, tax, and total amounts** to ensure correctness and detect inconsistencies.
- **Duplicate receipts are identified** by comparing cleaned OCR text, preventing repeated storage of the same receipt or invoice.

Store Structured Results in Database

- Validated receipt data is converted into a **structured format** and stored in an SQLite database.
- The database enables **efficient retrieval, history tracking, and analytics** for future processing and reporting.

Apply Regex + NLP to Parse Key Fields

After OCR, rawtext oftencontainsnoise andinconsistencies. Ourhybrid extractionapproachcombinesNatural Language Processing (NLP) and Regular Expressions(Regex) totransformthisinto structured data.

NLP-Based Extraction

- An NLP model (Gemini LLM) is used to understand the **semantic structure** of receipts.
- It identifies fields such as:
- Vendor name
- Date
- Line items (item name, quantity, price)
- Subtotal, tax, and total
- NLP provides flexibility to handle different receipt layouts and formats.

Regex-Based Extraction

- Regex patterns are applied as a **deterministic fallback** to extract critical numerical fields.
- Regex is used to:
- Extract dates in common formats
- Identify monetary values near keywords like *Total*, *Tax*, *Subtotal*
- Capture invoice or receipt IDs
- Regex ensures consistent extraction when NLP output is incomplete or uncer tain.

Hybrid Strategy Benefits

- NLP handles structure and semantics.
- Regex ensures accuracy and consistency.
- This hybrid approach improves robustness across varied receipt formats.

Validate Totals and Detect Duplicates

Raw extracted values often contain inconsistencies. This critical validation layer ensures data accuracy and integrity before storage.

Total & Tax Validation

- Verifies financial correctness: *Subtotal*+*Tax* ≈ *Total*.
- Applies tolerance for rounding errors.
- Prioritizes printed totals over computed values.
- Validates tax percentages within reasonable ranges.

Date & Required Field Validation

- Normalizes dates to `YYYY-MM-DD` format.
- Distinguishes and flags missing or invalid date formats.
- Ensures mandatory fields (vendor, date, total) are present.
- Missing fields are flagged for review.

Duplicate Detection

- Duplicate receipts are detected by comparing cleaned OCR text against stored records.
- If a receipt with the same textual fingerprint already exists:

1. The system prevents re-insertion
2. A warning is displayed to the user

- This avoids duplicate financial records and maintains data integrity.

Store Structured Results in Database

Once extraction and validation are complete, the verified data is stored in persistent storage using SQLite, ensuring data integrity and accessibility.

Database Structure

- Receipts are stored in a structured format:
  - Receipt ID
  - Vendor name
  - Date
  - Subtotal, tax, total
- Line items are stored separately and linked using a foreign key relationship.

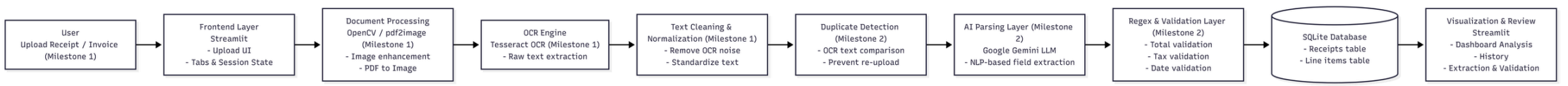
Benefits of Structured Storage

- Enables fast searching and filtering
- Supports analytics and reporting
- Allows historical tracking of receipts
- Prevents data loss and duplication

Persistence Strategy

- Only validated and non-duplicate receipts are stored.
- Structured storage ensures the system can scale to handle large numbers of receipts reliably.

Detailed System Architecture



Thank You