Fagarita • Servandil • Magno • Catanus

USER MANUAL

ExCeipt: Automated Bookkeeping Through Improved Information Extraction

EXCEIPT: AUTOMATED BOOKKEEPING ENCODING THROUGH IMPROVED INFORMATION EXTRACTION ON RECEIPTS

An Undergraduate Thesis

Presented to the Faculty of the

College of Information and Communications Technology

West Visayas State University

La Paz, Iloilo City

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Computer Science

by

Dave F. Fagarita

Jimuel S. Servandil

Jannica Mae G. Magno

Jeziah Lois C. Catanus

June 2024

Disclaimer

This software project and its corresponding documentation entitled “ExCeipt: Automated Bookkeeping Encoding through Improved Information Extraction on Receipts” is submitted to the College of Information and Communications Technology, West Visayas State University, in partial fulfillment of the requirements for the degree, Bachelor of Science in Computer Science. It is the product of our own work, except for the utilization of the LayoutLMv3 model on HuggingFace Transformers.

We hereby grant the College of Information and Communications Technology permission to freely use, publish in local or international journal/conferences, reproduce, or distribute publicly the paper and electronic copies of this software project and its corresponding documentation in whole or in part, provided that we are acknowledged.

Dave F. Fagarita
Jimuel S. Servandil
Jannica Mae G. Magno
Jeziah Lois C. Catanus

June 2024

Table of Content

Getting Started

Introduction

Efficient bookkeeping is essential for business operations, but manually encoding receipts can be time-consuming and error-prone. This project aims to automate bookkeeping by developing a web application that extracts information from receipts. The system scans an image to determine if it is a receipt. If confirmed, it uses the LayoutLMv3 model to automatically extract and categorize the text. The extracted information is then saved to a CSV file for further analysis and record-keeping.

Installation

Prerequisites

- **Git**: Ensure you have Git installed on your machine. You can download it from here: https://git-scm.com/downloads
- **Git LFS**: Ensure Git LFS (Large File Storage) is installed. You can download it from here: https://git-lfs.com/
- **Python 3.11.4**: Make sure you have Python 3.11.4 installed. You can download it from here: https://www.python.org/downloads/release/python-3114/
- **Conda**: If you prefer using Conda for environment management, ensure it is installed. You can download it from here: https://www.anaconda.com/download
- **OCR.Space API key**: Create your free API key here: https://ocr.space/ocrapi

Steps to Set Up the Project

1. Clone the Repository

```
git clone https://github.com/Scezui/ExCeipt_WebApp.git
cd ExCeipt_Webapp
```

2. Pull LFS files

```
git lfs pull
```

3. Create a new environment

Using *conda*:

```
conda create --name myenv python=3.11.4
conda activate myenv
```

Using *venv*:

```
python -m venv myenv
```

```
source myenv/bin/activate #On Windows, use `myenv\Scripts\activate`
```

4. Navigate to the source directory

```
cd src
```

5. Install Dependencies

```
pip install -r requirements.txt
```

6. Add API key

   Enter the API key you created from OCR.Space

```
 python create_api.py
```

7. Run the web application

```
flask run
```

Or you can access a working web application here: https://innovex-exceipt.hf.space/

System Requirements

- Minimum system requirements

  Operating System: Windows 10 64-bit: Home or Pro (build 19041 or later), Enterprise or Education (build 18363 or later). windows 11 64-bit: Home, Pro, Enterprise, or Education versions.

  Processor: 64-bit processor with Second Level Address Translation (SLAT)

  GPU: GTX1050

  Memory: 4.00GB RAM

- Recommended specs:

  Processor: Intel Core i5-2400s @ 2.5 GHz or AMD FX-6350 @ 3.9 GHz or equivalent

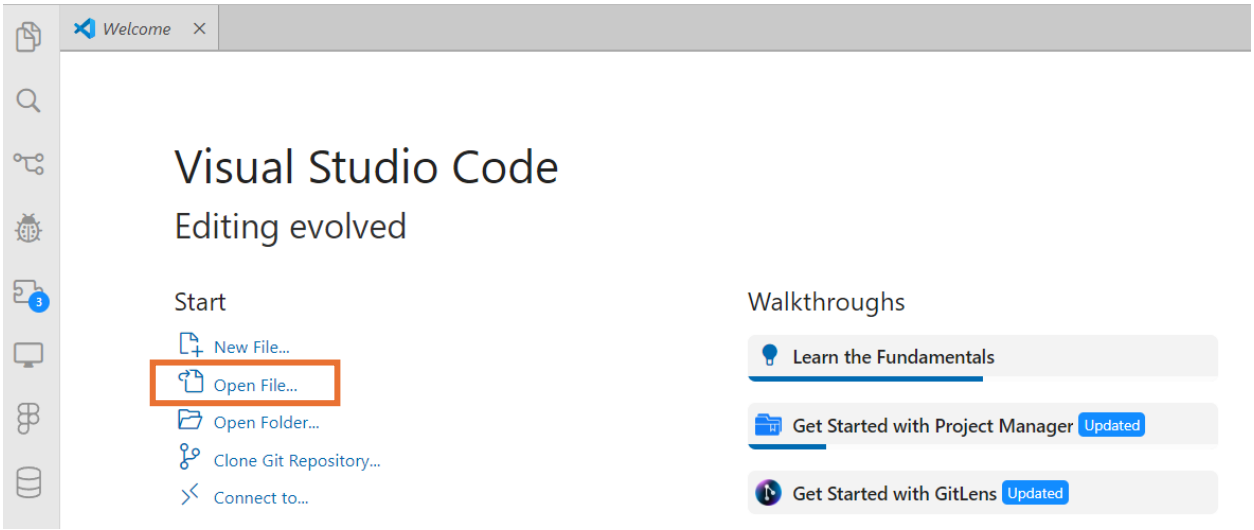
  GPU: GTX1650

  Memory: 8.00GB RAM or higher

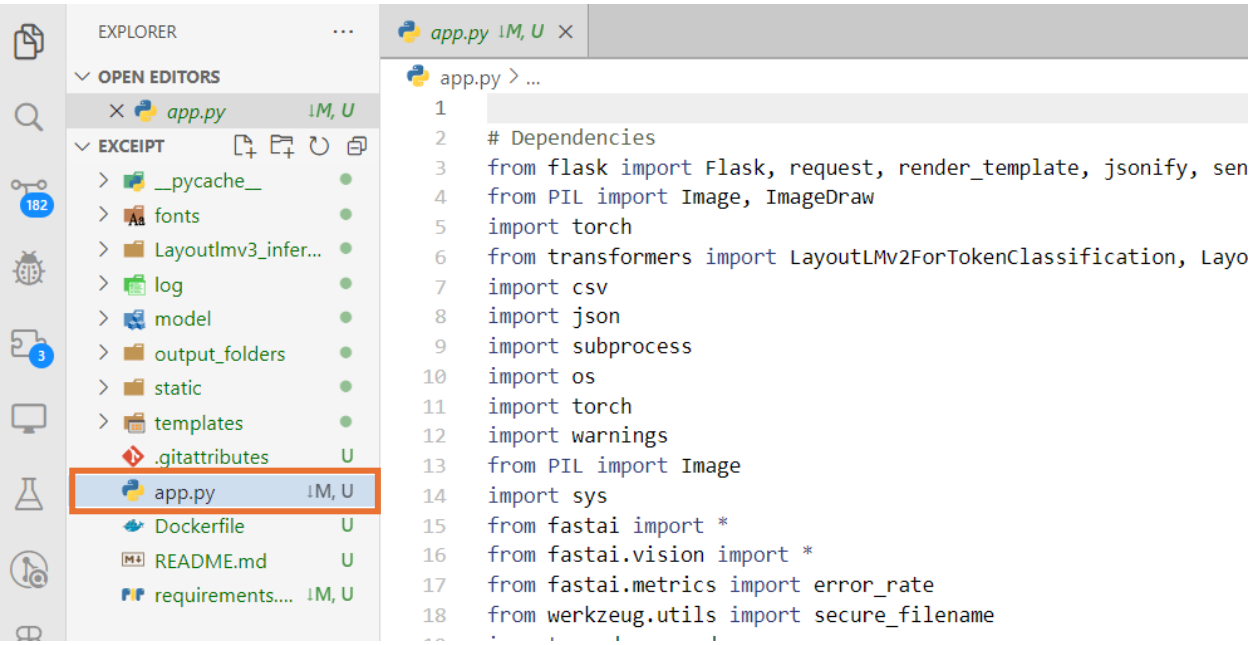
Usage

Step 1

Open Visual Studio Code or your preferred Integrated Development Environment (IDE). Click on the "Open Folder" option and locate the project's folder.



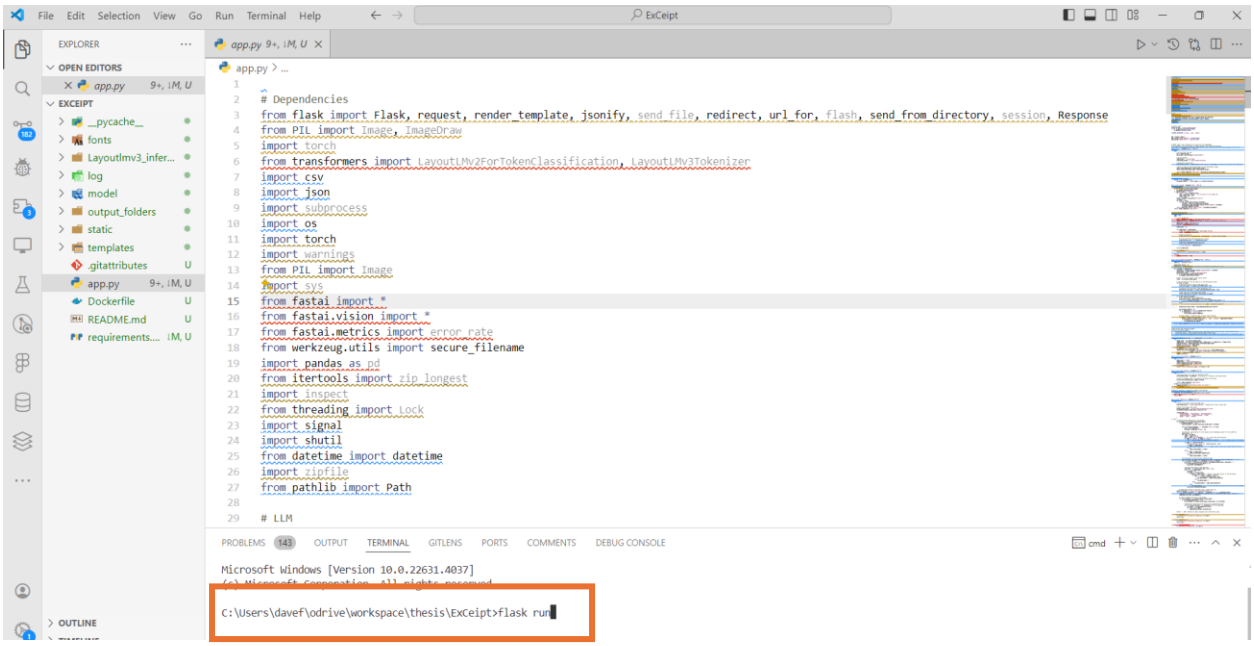
In the left-side panel of VS Code, find the file named app.py and open it.



Step 2

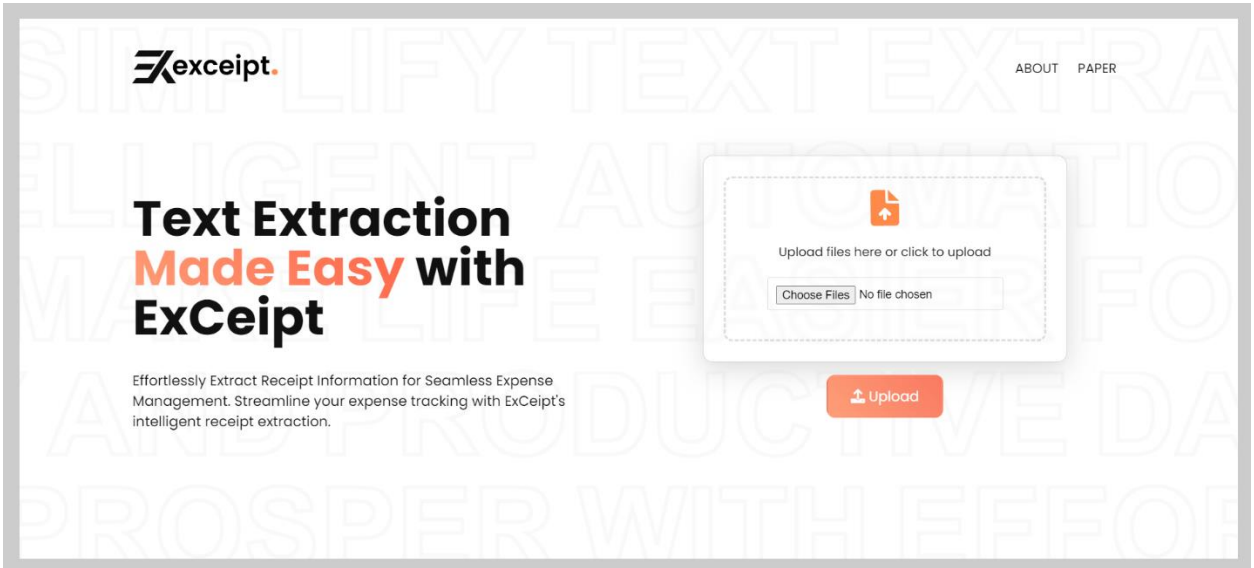
Toggle the built-in terminal in Visual Studio Code, then input the following command:

flask run



Step 3

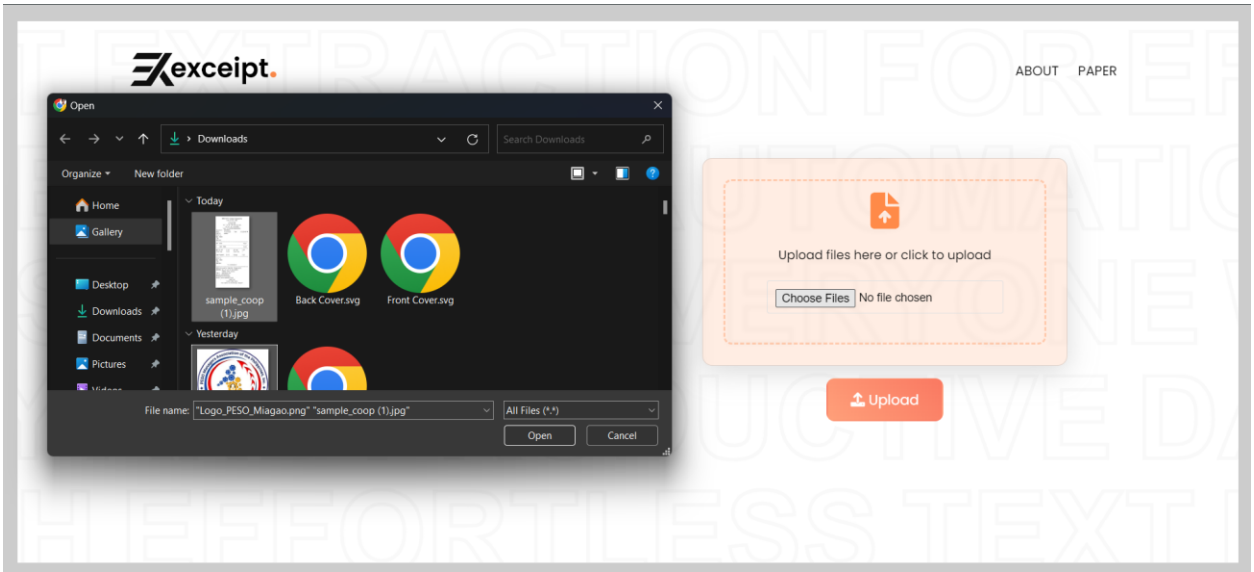
You will be directed to your browser, where the system's landing page will be displayed.

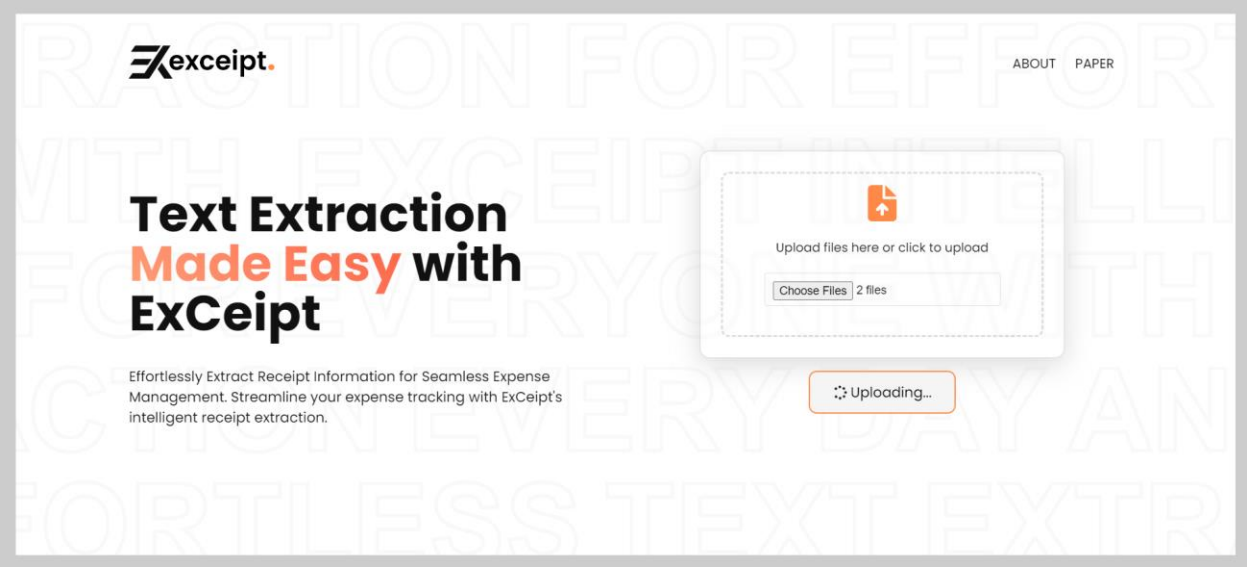


Step 4

Upload a scanned image of a receipt and submit it to validate whether the image is a receipt.

Note: The system supports multiple image uploads for validation and extraction.

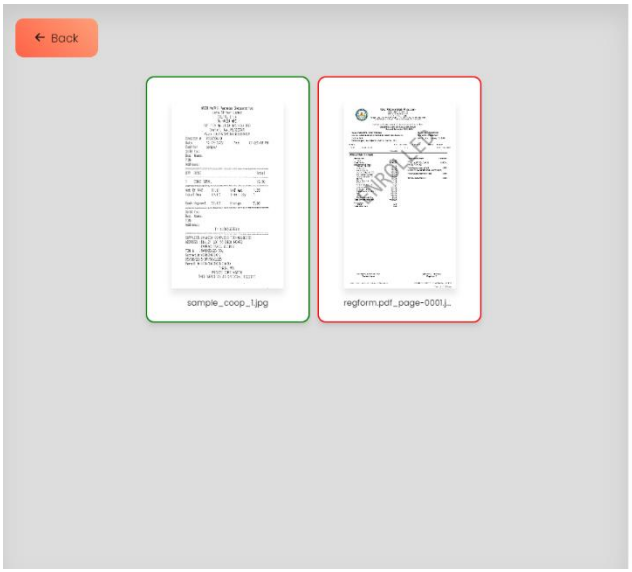


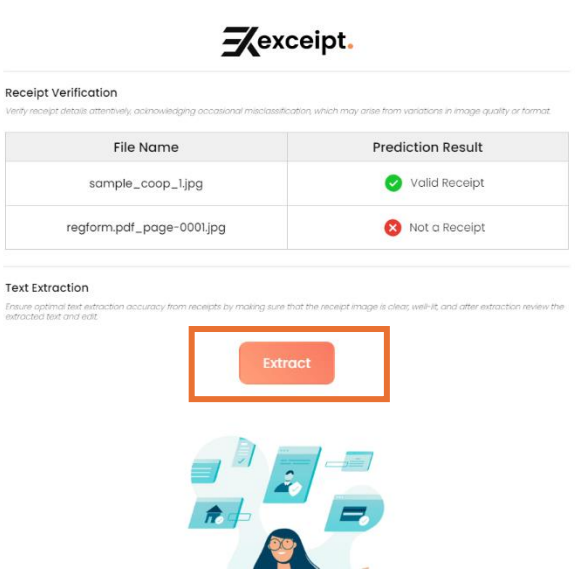


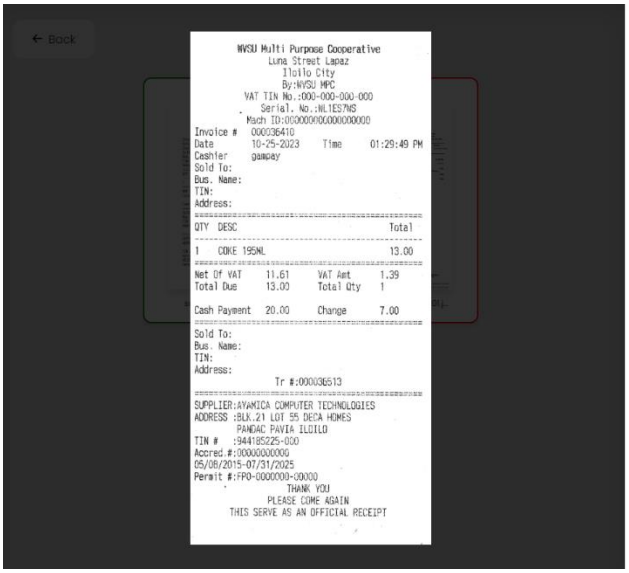
Step 5

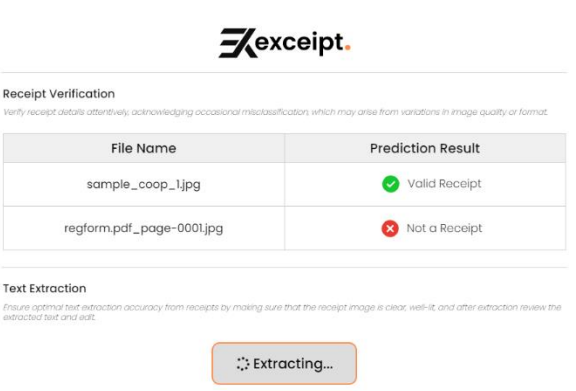
Once the image is submitted, the system will validate it and predict the result. After validation, proceed to the extraction process by clicking the "Extract" button.

Note: For batch image processing, invalid images will be automatically omitted after clicking the "Extract" button.



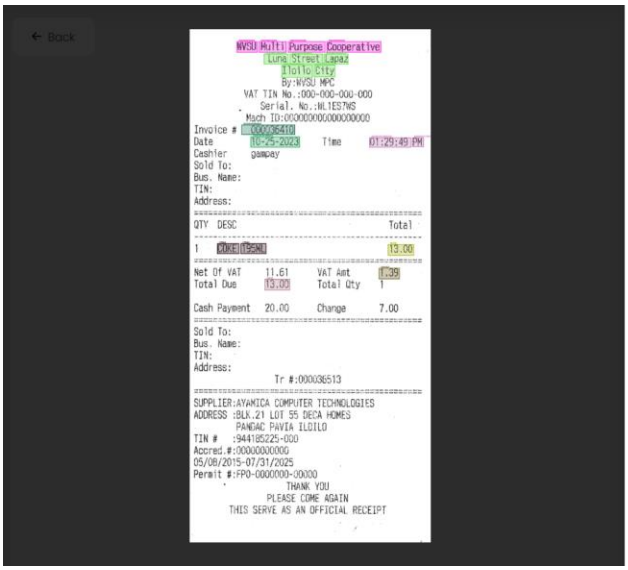


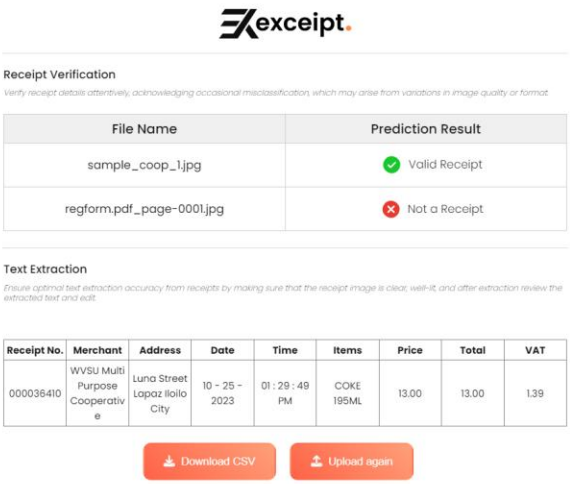




Step 6

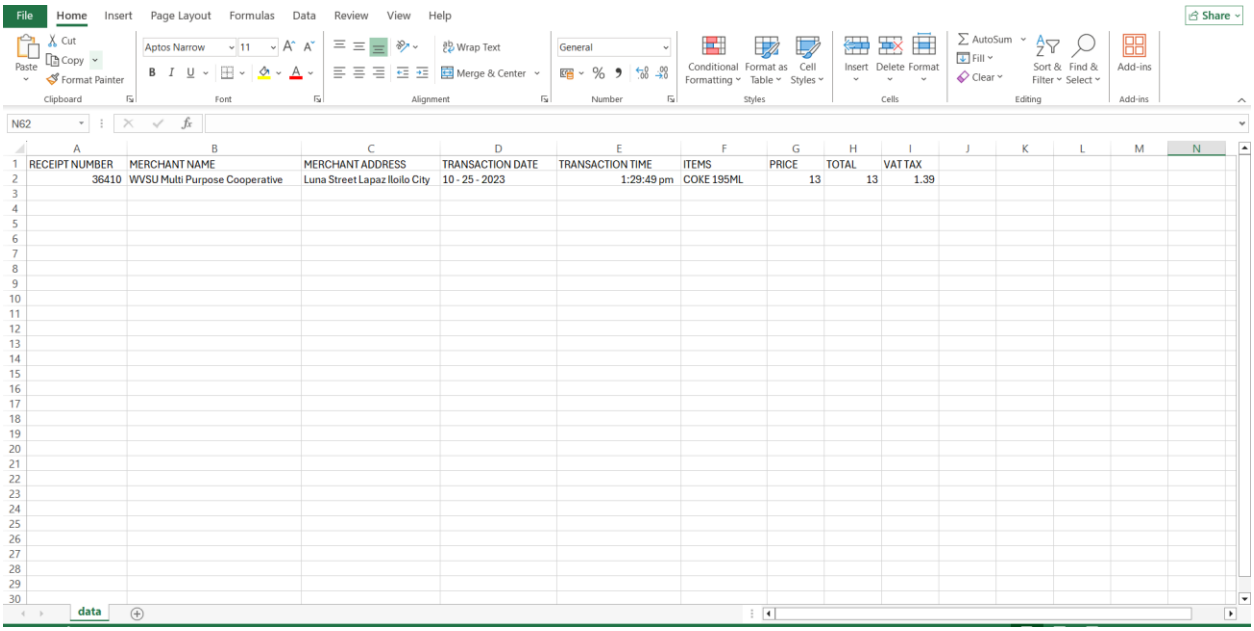
After extraction, the results will be displayed in a tabular format, allowing you to edit any incorrectly extracted values. The receipt image will also be annotated to indicate where the data was extracted from.





Step 7

The extracted results can be downloaded as a comma-separated values (CSV) file.



Troubleshooting

This section provides solutions to common issues you may encounter while using the ExCeipt system. If you experience a problem that is not listed here, please contact technical support for further assistance.

Troubleshooting for Manual Setup of the Standalone Web Application

If you are setting up the ExCeipt standalone web app manually, the following troubleshooting tips may help resolve common setup issues:

- **Issue: Unable to install dependencies using `pip` or `npm`.**

  Possible Causes:
  - Your environment may not be correctly configured, or there could be missing or outdated dependencies.
  - There may be a network issue preventing the download of packages.

  Solutions:
  - Ensure you have the latest version of Python and Node.js installed.
  - Run pip install -r requirements.txt for Python dependencies and npm install for Node.js dependencies.
  - Check your internet connection and try installing the dependencies again.
  - If issues persist, try running the commands in a virtual environment (e.g., `venv` for Python).

- **Issue: The Flask server fails to start.**

  Possible Causes:
  - There may be an issue with the configuration files (e.g., config.py) or missing environment variables.
  - A port conflict may be preventing the server from starting.

  Solutions:
  - Check the Flask app configuration in config.py and ensure all necessary environment variables are set.
  - Ensure that the port specified in your configuration is not being used by another application. You can change the port by setting the FLASK_RUN_PORT environment variable.
  - Run the server with python app.py and check for error messages to diagnose the issue.

- **Issue: Static files (CSS, JS) are not loading properly.**

  Possible Causes:
  - The static files may not be correctly linked in your HTML templates.
  - There may be an issue with the web server configuration, such as incorrect paths or permissions.

Solutions:

- Check the paths to your static files in your HTML templates and ensure they are correct.
- Ensure that the static folder is correctly configured in your Flask app and that the files have the correct permissions.
- Clear your browser cache and reload the page to see if the static files load correctly.

Issue: The web app shows a "500 Internal Server Error".

Possible Causes:

- There may be an issue with the application code, such as a syntax error or an unhandled exception.
- There may be an issue with the configuration or environment variables.

Solutions:

- Check the server logs for detailed error messages and traceback to identify the issue.
- Ensure all environment variables are correctly set and that the configuration file (config.py) is correct.
- If you recently made changes to the code, revert to the previous working state to isolate the issue.

FAQ (Frequently Asked Questions)

What is ExCeipt?

ExCeipt is an automated bookkeeping system designed to extract structured data from receipts using advanced machine learning techniques. It utilizes the LayoutLMv3 model for document layout analysis and information extraction, as well as a Convolutional Neural Network (CNN) for receipt validation. This system aims to streamline the bookkeeping process by automating the encoding of receipts, thereby saving time and reducing errors.

What problem does ExCeipt aim to solve?

ExCeipt addresses the challenge of manually encoding receipts, which is often time-consuming and prone to errors. By automating the extraction of key information from receipts, the system improves the efficiency and accuracy of bookkeeping, allowing bookkeepers and small to medium-sized businesses to focus on other critical tasks.

How does ExCeipt validate whether an image is a receipt?

ExCeipt uses a Convolutional Neural Network (CNN) algorithm to validate whether the uploaded image is a receipt or not. The CNN model analyzes the image to ensure it is suitable for further processing and information extraction.

What technologies are used in the development of ExCeipt?

ExCeipt is developed using several key technologies:

- **Optical Character Recognition (OCR):** Used to convert text from receipt images into machine-readable code.
- **LayoutLMv3:** A deep learning model used for document layout analysis and information extraction.
- **CNN:** Used for validating receipt images.
- **Flask:** The backend framework used to develop the web application.

What kind of receipts does ExCeipt support?

The system supports English-language receipts that include detailed lists of items purchased, their quantities, prices, and transaction dates. The receipts used in the study were collected from establishments like 7-Eleven, Iloilo Grace Pharmacy, and the WVSU Cooperative.

What are the specific outputs generated by ExCeipt?

The outputs generated by ExCeipt include structured data extracted from receipts, such as the receipt number, merchant name, merchant address, transaction date and time, item descriptions, prices, total amount, and VAT tax. This data is categorized and saved in a CSV file format.

How accurate is the ExCeipt system?

During testing, the LayoutLMv3 model achieved a 99.12% accuracy rate and a 95.35% F1 score in extracting and labeling receipt information. The CNN model for receipt validation attained a 99.77% accuracy rate. The system has been evaluated using ISO/IEC 25010 standards, demonstrating high quality and effectiveness across all relevant categories.

Who can benefit from using ExCeipt?

ExCeipt is particularly beneficial for bookkeepers, small to medium-sized enterprises (SMEs), and stakeholders in accounting and finance domains. It helps in reducing labor costs, improving data accuracy, ensuring compliance, and enhancing the overall efficiency of financial processes.

What are the limitations of ExCeipt?

The study focused solely on English-language receipts and on the primary step of bookkeeping, which is recordkeeping. It does not support multilingual receipts or other aspects of the bookkeeping process. Additionally, the data used in the system is anonymized to comply with data privacy laws.

What future enhancements are proposed for ExCeipt?

Future enhancements may include integrating more advanced Optical Character Recognition (OCR) tools, expanding the system to support receipts in multiple languages, and refining the algorithm for even greater accuracy and efficiency. There may also be further improvements in the LayoutLMv3 model and CNN for better performance.

Contacts

For any questions, feedback, or issues related to the ExCeipt system, please feel free to contact the research team directly. The researchers are available to provide support, clarify information, and assist with any technical difficulties you may encounter.

You can reach the researchers at the following contact details:

- **Dave F. Fagarita**
  dave.fagarita@wvsu.edu.ph
  +639934487291
- **Jimuel S. Servandil**
  Jimuel.servandil@wvsu.edu.ph
  +639275693028
- **Jannica Mae Magno**
  jannicamae.magno@wvsu.edu.ph
  +639706460799
- **Jeziah Lois Catanus**
  jeziahlois.catanus@wvsu.edu.ph
  +639517388236