CMPT 310 Project Proposal - Group 6

1. **Group Information**
    a. Farzan Ustad, 301576889, fua@sfu.ca
    b. Andy Wang, 301585112, rwa148@sfu.ca
    c. Daniel Shi, 301594914, dsa179@sfu.ca

2. **Project Idea**

    We will build a ML based receipt scanner that extracts key financial information from images of physical receipts and logs it into an organized spreadsheet. The goal is to automate expense tracking for individuals, students, or small organizations by removing the need for manual data entry. This system addresses the common problem of lost or untracked purchases due to lack of user time, enabling users to digitize their expenses and maintain accurate records with ease.

    The system will take an image of a receipt as input, either uploaded manually or captured via camera. It will output a structured entry in a spreadsheet containing fields such as date, merchant name, total amount, tax, and payment method (if available). Our AI approach will rely on Optical Character Recognition (OCR) using tools such as Tesseract or Google Vision API to convert receipt text to raw data. We will then use rule-based pattern matching (e.g., regular expressions and keyword filtering) to extract relevant fields. If time permits, we may integrate a basic classifier to categorize expenses (e.g., food, transportation, etc.) based on detected keywords.

3. **Tools and resources**

    pytesseract – OCR engine to extract text from receipt images.
    opencv-python – For image preprocessing (resizing, denoising, thresholding).
    re (regex module) – To extract specific fields like date, total, and vendor from text.
    pandas – To structure and organize extracted data into a table.
    openpyxl – To write data into an Excel (.xlsx) file.
    nltk or spacy – For basic NLP if you add categorization or cleanup of messy OCR output.
    gspread – (Optional) To integrate with Google Sheets instead of Excel.
    JaidedAi – For text extraction from images for different languages.
    - https://github.com/JaidedAI/EasyOCR

    Our model will train on data and images sets from Kaggle and other public libraries.

4. **Project plan/Timeline**

<u>Milestone 1 – July 2</u>

Goal: Basic working prototype with OCR and simple data extraction

- Collect sample datasets of receipt images from Kaggle and public datasets.

- Implement OCR functionality using pytesseract on a small subset of clean images.
- Apply OpenCV preprocessing (e.g., grayscale conversion, thresholding, noise reduction) to improve OCR accuracy.
- Optionally, train the model to recognize whether the image is a receipt or not.

Expected Deliverable: A minimal program that can successfully convert a clean receipt image into structured text.

Milestone 2 – July 30

Goal: Extended system with simple ui, categorization, and csv or Excel export

- Use regular expressions to extract key information such as:
    - Date
    - Merchant name
    - Total amount
- Make a simple text based UI
- Add support for batch processing multiple receipts.
- Output raw extracted fields into a CSV or DataFrame

Expected Deliverable: A more robust system that accurately extracts multiple fields from various receipt types and logs them in a structured Excel file, with basic categorization.

5. MVP

Our simplest working version of the system is a machine learning program that can use computer vision to identify text of clear receipts and convert it into CSV. The core functionality includes two parts, recognizing receipts and converting to texts and numbers, then, the algorithm will categorize the date, store name, total spent etc.