The East Midlands AI Project

PRESENTED BY **TEAM 3**:

- **HARMAN SAINI** - API RESEARCHER
- **VICKY NAROTAMO** - PYTHON DEVELOPER
- **KISHAN MANSUKLAL**- DATA CURATOR / ANALYST
- **KISHAN MANSUKLAL** - TESTER / DOCUMENT LEAD
- **THANAA SULUM** – PRESENTATION/DOCUMENTOR LEAD

DE MONTFORT UNIVERSITY – LEAD AND INSPIRE PROGRAMME

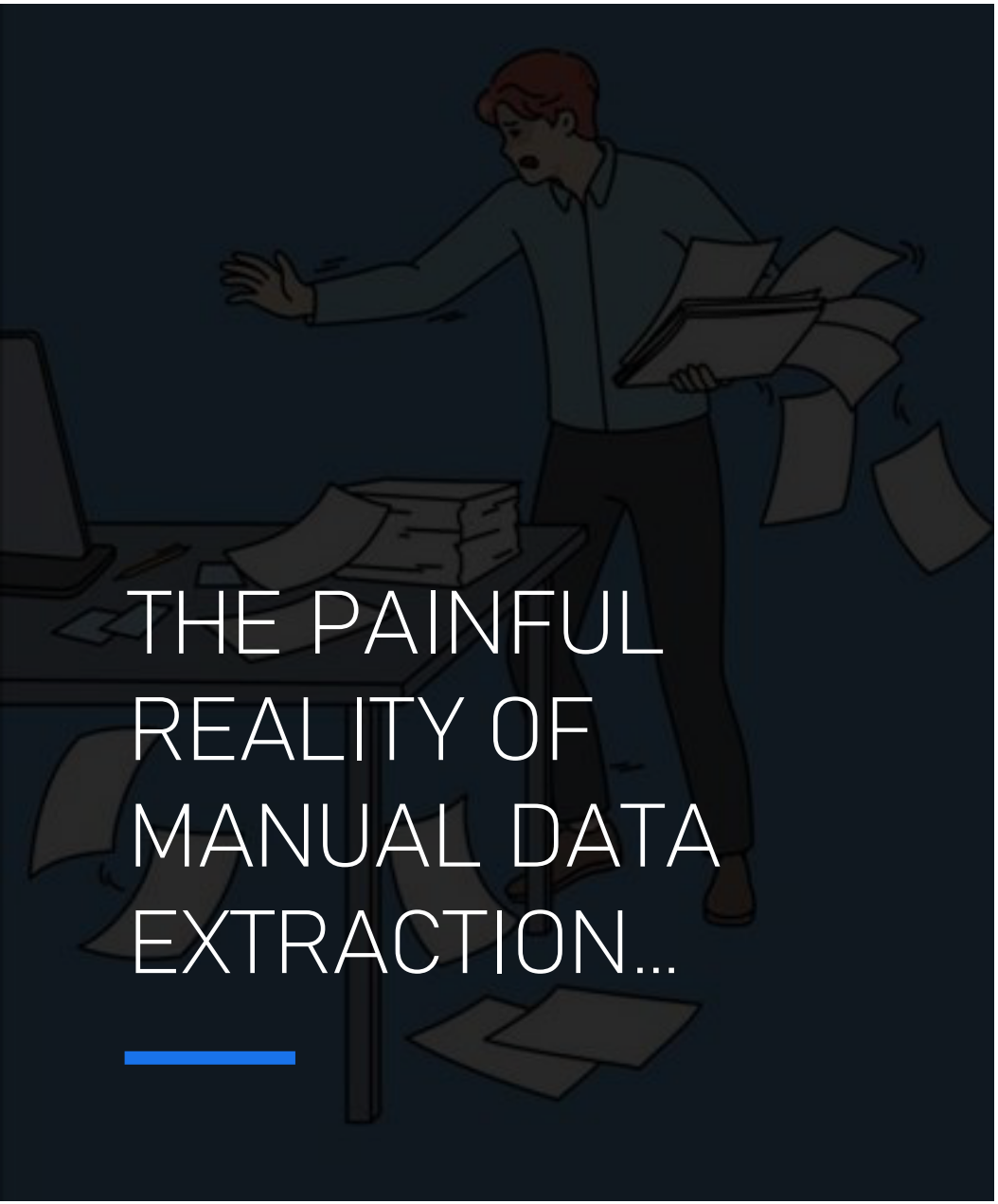
Project Overview:

- The purpose of the Lead and Inspire Programme: fostering innovation, developing leadership skills, and applying technology to real-world business challenges.
- **The Task Presented by:** A leading business that supports organisations by connecting, representing and championing businesses across the East Midlands, using their collective strengths to create opportunities and growth.
- **In the SIMPLEST TERMS, OUR task is...**
- **“To leverage the utilisation of AI within the business department to develop a new approach to reading PDFs and extracting information out of it”.**

Further detail of our assigned roles

Team Member Role Key Responsibilities:

- Vicky: Python Developer Scripted the full PDF ➜ Image ➜ Text pipeline
- Kishan: Quality Assurance + Tester Checked image clarity and output accuracy
- Harman: API Researcher + Helped validate Gemini setup and usage
- Thanaa: Presenter/Documenter Compiled findings + created the presentation



THE PAINFUL REALITY OF MANUAL DATA EXTRACTION...

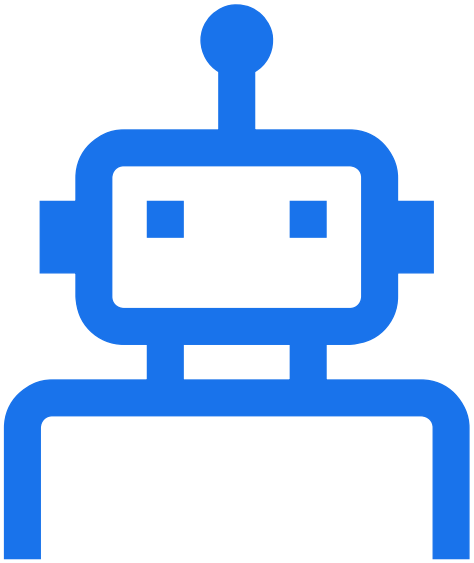
HERE'S A FEW STATS TO TAKE NOTE OF:

- FOR **10,000 DATA ENTRIES,** HUMANS WOULD COMMIT BETWEEN **100 AND 400 ERRORS.** (DOCUCLIPPER)
- PWC RESEARCH SUGGESTS THAT EVEN RUDIMENTARY AI-BASED EXTRACTION TECHNIQUES CAN SAVE BUSINESSES **30-40% OF THE HOURS** TYPICALLY SPENT ON SUCH PROCESSES. (PWC)
- INTELLIGENT DOCUMENT PROCESSING CAN ACHIEVE **500% TO 1000% PRODUCTIVITY GAINS** FOR SPECIALIZED TEAMS. (DEVOTEAM)
- ONE CASE STUDY OF IMPLEMENTING AN AI-POWERED OPTICAL CHARACTER RECOGNITION SYSTEM (A TYPE OF TECHNOLOGY THAT CONVERTS PDS TO MACHINE READABLE TEXT) FOR FORM FILLING ELIMINATED ERRORS AND ENABLED REAL-TIME DATA VALIDATION, RESULTING IN **IMPROVED DATA ACCURACY.** (HABILEDATA)
- A QUICK SEARCH ONLINE WAS ALL IT TOOK TO FIND PEOPLE SHARING THEIR AGONISING MANUAL DATA EXTRACTING EXPERIENCE, AND IT WASN'T PRETTY. A SMALL SURVEY WAS DONE ONLINE AND OVER **114 SPENT MORE THAN 50 HOURS** EXTRACTING DATA AMONGST THEMSELVES.

IT’S AI VERSUS THE HUMAN HAND

The world is now digital, embrace efficiency.
As great as we are as human beings, we lack utmost efficiency, we make mistakes, and sometimes these mistakes can be costly…

But we can fix that…this is what “IT” can do for you:



•Our AI-powered system is built with strong foundations due to the robust methodology designed to automate fast processing’s of information extraction. It:

- **Minimises human error** – improvement **of accuracy**, lessens the chance of any missing data or redundancy.
- Is **easy** to work with, as it's got a very simple interface.
- Overall, it **maximises efficiency**, and utilizes an effectively **modernised approach to problem solving**, such an adaptation would lead to **improvement to productivity**.

Our System's Core Capabilities : The Overview

- Here's how it works:
- **Convert PDFs to Images**: Using the pdf2image and Pillow libraries, each receipt PDF is split into image pages.
- **Extract Text with Gemini**: We send each image to Google's Gemini 2.0 Flash model, which reads the receipt layout and pulls out store names, dates, totals, item lists, and so on.
- **Save as Text Files**: We take the text Gemini pulls out and save it into plain .txt files that keep the same layout as the original receipts. That way, you can check them at a glance and then feed the data straight into whatever you need.
- **Tech Stack** We are running this on Python 3.11 and using the official Google Gen AI SDK for Python along with pdf2image, Pillow, python-dotenv, and tqdm for nice progress bars.

Continuation...

- **Setup Steps:** First, install all the required packages by running: pip install git+https://github.com/googleapis/python-genai.git pip install python-dotenv , df2image pillow ,tqdm
- "Poppler is a free and open-source software library used for rendering Portable Document Format (PDF) documents.It provides a way to display, extract, and manipulate PDF content"
- If you're on Windows, grab Poppler, extract It to C:\poppler, and add C:\poppler\bin to your PATH.
- "C:" - refers to the hard drive
- Bin – is where the executable files are kept, within a folder
- In the project folder, create a file called .env with this line in it: GOOGLE_API_KEY=your_actual_api_key_here
- Then just run python convert_all_pdfs.py to turn your receipts into PNGs and python Gemini_Integration.py to send them to Gemini and save out the text.
- **Prompt Design Tips** To make sure Gemini gives you exactly what you need, try this: ask it specifically for the store name, the date, the total amount, and the full list of items. If you can, show one or two examples of exactly how you want the output formatted. You can even ask for JSON if you're going to drop the results straight into a database. Finally, when you have a single-page receipt, put the image before any text instructions so Gemini focuses on the picture first.

- To achieve **scalability**:

- Our project is scalable, as it can be cloned and reused by other developers or teams. It supports any number of receipt PDFs and automatically processes them in bulk, making it suitable for small or large datasets.

- To achieve **sustainability**:

- Reusability: The codebase is modular and well-documented, making it easy to adapt for other types of documents – not just receipts. This increases the long-term value and use cases of the system.

- **Environmentally Efficient** (Digital over Manual) - By automating data extraction from receipts, the system reduces the need for manual typing, scanning, or paper-based storage. This contributes to more efficient, paperless workflows – a more sustainable data management approach.
- **Minimal Dependencies** - The project relies only on lightweight, open-source libraries like pdf2image, Pillow, and google-generative ai, which helps ensure low resource usage and long-term maintainability.
- **Team Knowledge** - Sharing The use of GitHub and clear documentation allows other team members or future contributors to pick up and improve the system, encouraging collaborative sustainability. 5. Extensibility Additional features (e.g., auto-summarization, data analysis, or translation) can be easily added without restructuring the project, making it future-proof.

Scalability & Efficiency

- **Python:**
- Pushes for **scalability** as Python is one of the easy coding languages, so it’d be easy to maintain into the future.
- **Readability of code** and the simplicity of it ensures that users aren’t struggling, even as the system grows, due to the addition of features.
- Leads to the **vast libraries and frameworks** that Python offers, like the ones we’ve used, “tqdm” for progress bars for example.
- Available modules cushion the path to a **faster development cycle**, which results in further scalability.
- Python also works on many operating systems, so it's easily a **universal programme**, that can be used anywhere.

- **Gemini AI:**
- **Cost-efficient:** The number of receipts will not cause the servers to crash as its infrastructure was made to manage that load.
- **Security:** Touch ups and updates are done by Google themselves, so it’s mostly hands-off.
- **Flexibility:** State-of-the-art AI for text extraction, as its cloud-based it is also scalable and reliable.
- It can also **manage varied layouts**, though receipts may not look the same, it causes no trouble for Gemini AI.

Tools & Technologies Used

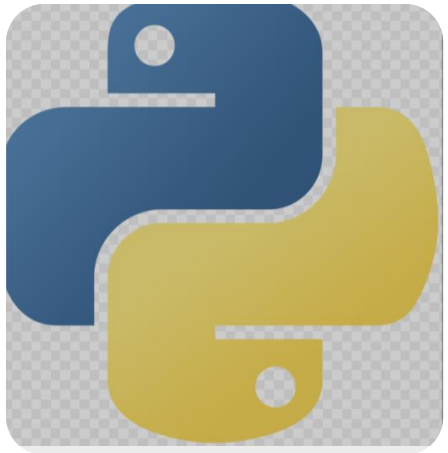
- The initial step involves **integrating Gemini with PyCharm**. Following this, the process would entail **gathering PDF files, converting them into images, and then extracting text** from those images.

- **Python:** The programming language used.
- **google-genai:** For integrating with Google Gemini AI.
- **pdf2image + Pillow:** Libraries for converting PDFs to images.
- **Poppler for Windows:** An essential dependency for pdf2image on Windows.
- **.env + python-dotenv:** For securely managing API keys.
- **TQDM (progress bar):** Provides visual progress feedback.
- **GitHub:** For version control and collaborative development.

Key Resources provided to us:

Gemini AI Image to Text example: https://ai.google.dev/gemini-api/docs/text-generation?lang=python
Gemini AI documentation: https://ai.google.dev/gemini-api/docs/quickstart?lang=python
Install Python and PyCharm: https://youtu.be/lWekkVSXRsU
PDF to Image library: https://pypi.org/project/pdf2image/

All above documentation should be enough to perform the task.



THE METHODOLOGY. THE "HOW".

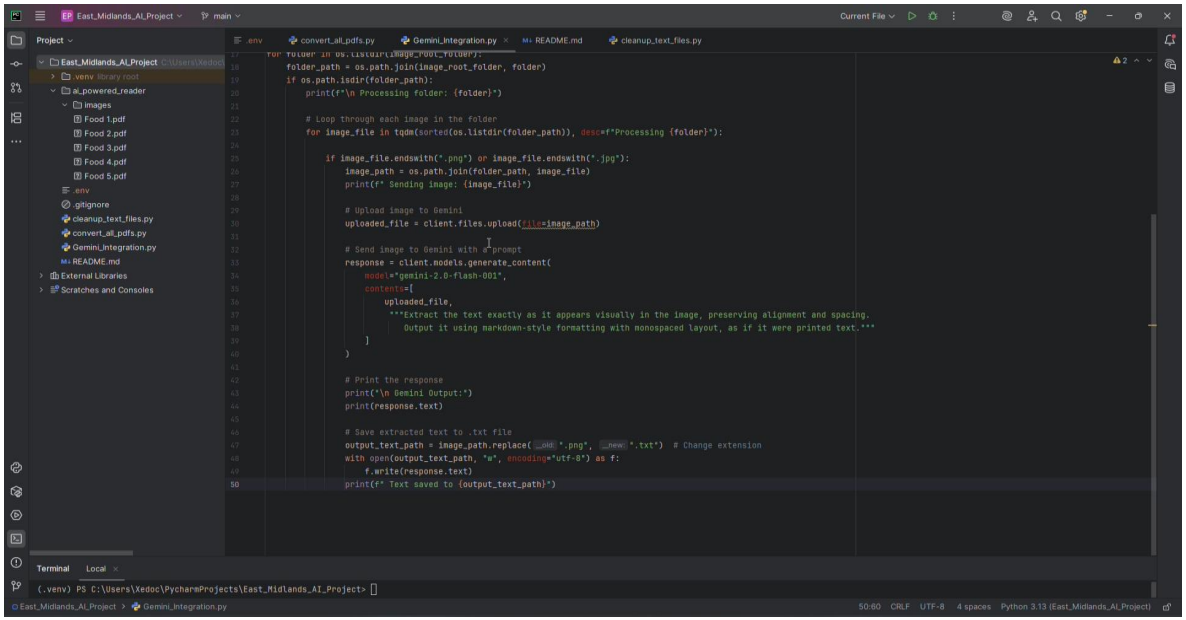
A SYSTEMATIC APPROACH TO DATA EXTRACTION

Our systematic approach involved identifying the problem, thorough research and planning, clear role assignment, and rigorous testing – all leading to the finished product.

- You may see a demonstration in the next slide.

Walkthrough

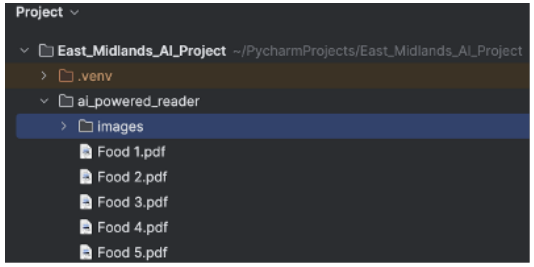
- A a quick video demonstration of our final product, by the lead developer :



HOW DOES IT DO THAT??...

PHASE 1: DATA ACQUISITION & PRE-PROCESSING:

(An image of an in-tray)



(ai_powered_reader folder)

The "In-Tray" (Input):

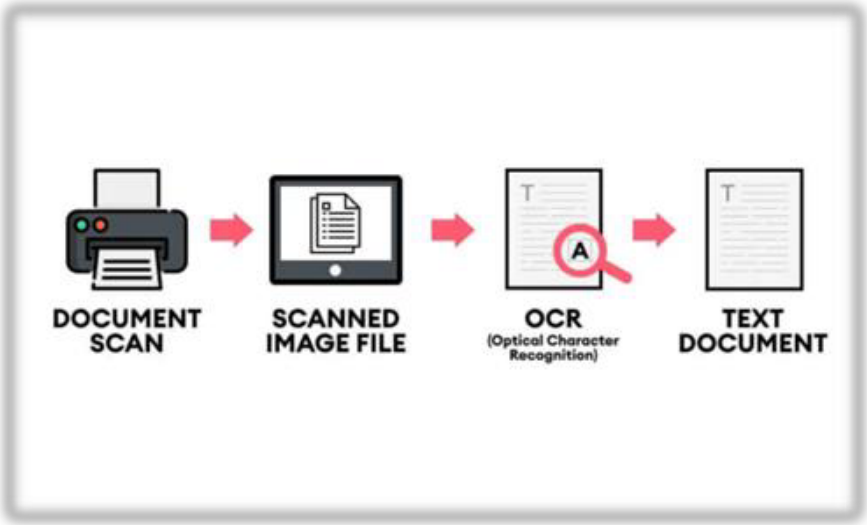
First, the user would start by simply placing the scanned PDF receipts into the designated input folder, “ai_powered_reader”, within the programme – this acts as the system's digital in-tray .

```
Saved ai_powered_reader/images/Food 1/page_1.png
Saved ai_powered_reader/images/Food 2/page_1.png
Saved ai_powered_reader/images/Food 2/page_2.png
Saved ai_powered_reader/images/Food 3/page_1.png
Saved ai_powered_reader/images/Food 3/page_2.png
Saved ai_powered_reader/images/Food 3/page_3.png
Saved ai_powered_reader/images/Food 4/page_1.png
Saved ai_powered_reader/images/Food 4/page_2.png
Saved ai_powered_reader/images/Food 5/page_1.png
```

A pleasant and convenient message then appears on screen, which indicates the progress and success being converted into an image.

- **The "Visualizer"** : The first of automated processes. Run the “**convert_all_pdfs.py**” -> a python file this receives the pdf, after it searches for the pdfs, which will eventually create a separate **high quality image file (PNG).**

Phase 2: AI-Powered Text Extraction (OCR):



- The "Intelligent Reader" (**Gemini AI** & Gemini_Integration.py): Moving on, the user would “run the program”, this time it’s the python “Gemini_Integration.py”, the script from this program would find its way into the folder with the freshly converted images, and send it off to Googles powerful **Gemini 2.0 Flash AI model**.
- Here, Gemini AI now processes the images based on the prompt in the same script, it acts as an Optical Character Recognition (OCR) engine, as an aspect of Gemini AI, meticulously “reading” the contents of the image, and accurately transcribing it into digital text.
- OCR itself is the technology that converts images of text into machine-coded text, making non-searchable images readable by computers.

Unit 3-5 St Martin's Square
Leicester
LE1 5DF
Tel No:0116 251 9999

CHK# 39 TBL# 34
Silky # 2233
18/01/2024 18:12:46 GUESTS 2

1 Thai Green Curry	13.50
Vegetable	
1 Chilli & Basil Gra Pao	12.95
Vegetable	
1 Chilli Mocktail	6.25
1 Trip CBD Peach Ginger	4.50
1 Tom Yum Fried Rice	4.75
SUBTOTAL:	41.95
Service Charge	4.20
TOTAL:	£46.15

Sales-VAT	VAT Rate	VAT Amt	Total
34.96	20.00%	6.99	41.95

Last Serviced
18/01/2024 19:14:07

A discretionary 10% service charge
has been added to your bill
Thank you and please come again soon
VAT: 797 816 558
www.gigglingsquid.com

For dine-in customers, a discretionary
10% service charge has been
added to your bill.

Thank you, please come again soon.

How did we do today?
We would love to hear your feedback.
Share your thoughts with us and you
could win a £100 Giggling Squid
voucher.

Visit:
www.gigglingsquid.com/feedback

VAT : 797 816 558
www.gigglingsquid.com

Gemini Output:

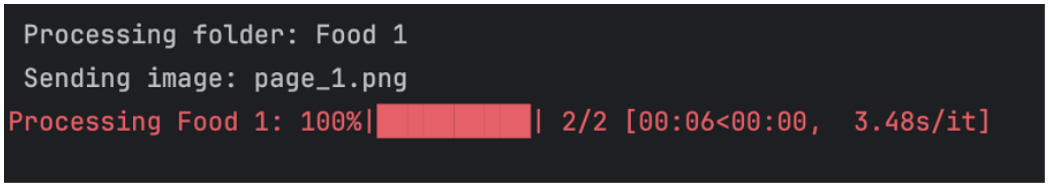
```
Giggling Squid
Unit 3-5 St Martin's Square
Leicester
LE1 5DF
Tel No:0116 251 9999
-------------------------------------------
CHK# 39          TBL#          34
Silky            # 2233
18/01/2024 18:12:46   GUESTS          2
-------------------------------------------
1 Thai Green Curry                13.50
  Vegetable
1 Chilli & Basil Gra Pao          12.95
  Vegetable
1 Chilli Mocktail                  6.25
1 Trip CBD Peach Ginger            4.50
1 Tom Yum Fried Rice               4.75
SUBTOTAL:                         41.95
Service Charge                     4.20
TOTAL:                           £46.15
-------------------------------------------
Sales-VAT VAT Rate    VAT Amt      Total
34.96     20.00%      6.99       41.95
-------------------------------------------
Last Serviced
18/01/2024 19:14:07
-------------------------------------------

A discretionary 10% service charge
has been added to your bill
Thank you and please come again soon
VAT: 797 316 558
www.gigglingsquid.com
```

Gemini's Text extraction Result (transcribed receipt):

- **Along with it, the terminal window below would display all its progress for the user, with a small reassuring progress bar, showing the percentage of the completion of each image being read.**

  Now, the user waits until it’s done it all…but not for long, since the process of it all, is amazingly and efficiently rapid.

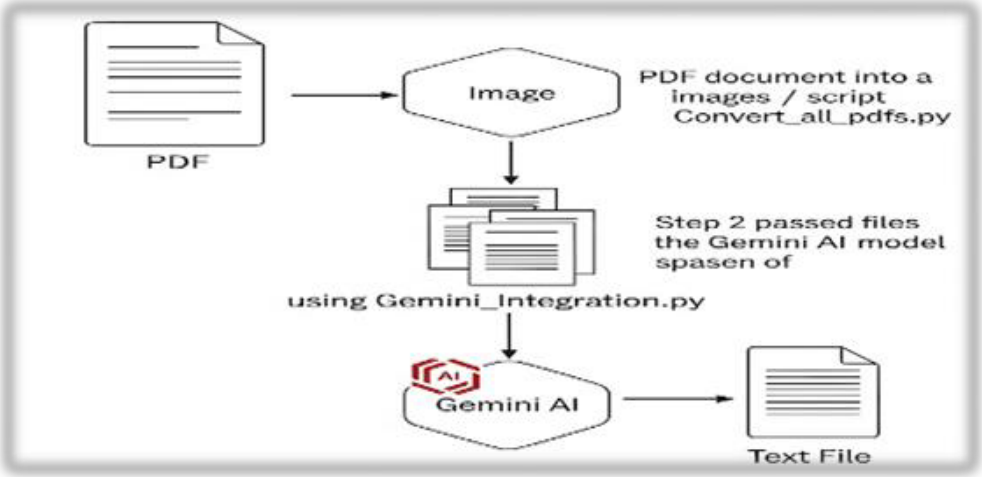


THIS IS A PROGRESS BAR.

"TQDM IS A PYTHON LIBRARY THAT PROVIDES FAST, EXTENSIBLE PROGRESS BARS FOR LOOPS AND ITERABLES"

(FUN FACT “TQDM” IS AN ABBREVIATION OF "PROGRESS" IN ARABIC FROM THE WORD “TAQADDUM”):

Phase 3: Data Storage & Accessibility:



- Finally, the well awaited results would have been displayed right before your eyes. Raw text that has been extracted successfully. The data's initial format is even persevered, in a layout that is concise and neat, that it's easy to identify each letter and number.

- As for the finishing touches, the extracted data is saved into a new text file, automatically. This provides you with a convenient digital and searchable record of the contents.

```
``
Text saved to ai_powered_reader/images/Food 1/page_1.txt
```

- An image of a saved .txt file:

Then, we began we began to question...

From Data to Insights:

- We had questions, to test if it was really regurgitating the correct information, for example:

- **What are the common items ordered from the 5 receipts?**

- **Can you identify price trends or high-value items from the receipts?**

- **Can you suggest any business insights based on multiple receipts?**

And this is what we got. Accuracy.

Common Purchases & Pricing Overview

Common Items Ordered:

Based on the 5 receipts, the most frequently purchased items fall into these categories:

1. **Drinks** (Coffee, Juices, Smoothies): ***9 items total*** (e.g., Espresso, Innocent Smoothie, Moju Ginger Shot, Lucozade, Mocktails).
   - *Insight:* Consistent demand for both hot and cold beverages, including health-conscious choices.
2. **Wraps / Savoury Baked Goods**: ***7 items total*** (e.g., Falafel & Houm Wrap, Spicy Bean Wrap, Cheese & Jalapeno Pasty).
   - *Insight:* Strong preference for convenient, often vegetarian-friendly, grab-and-go lunch options.
3. **Snacks/Sides**: **7 items total** (e.g., Pringles, Chilli Crisps, Double Choc Brownie Bites, Fruit Flapjack).

Price Trends & High-Value Items:

1. **Highest Individual Item Prices**: ***Found in restaurant meals*** (e.g., Giggling Squid: Thai Green Curry **£13.50**, Chilli & Basil Gra Pao **£12.95**).
2. Grab & Go Pricing: Individual items from supermarkets/cafes are typically under £5 (e.g., Wraps £2.75 - £4.25, Coffees £2.10 - £2.95).
3. Meal Deal Discounts: Frequently observed promotions (Tesco, Sainsbury's, M&S, Pret) reduce total costs by £1.10 - £2.45, bundling items for a set price (e.g., M&S Meal Deal for £5). This indicates a key sales strategy.
4. VAT: Most items are subject to a standard 20% VAT, though some food items can be zero-rated.

Key Business Insights

Strategic Focus Areas:

- Analysis of multiple receipts reveals several consistent business strategies:
- **Dominance of Convenience:** Most purchases are from express stores, cafes, and quick-service outlets, highlighting a strong market for on-the-go meals, snacks, and drinks.
- **Leveraging Promotions:** Extensive use of "Meal Deals" and other discounts is a critical sales tactic to increase average transaction values and offer perceived customer savings.**C**
- **customer Engagement & Loyalty:** Businesses actively encourage feedback (via websites/surveys) and promote loyalty programs (Clubcard, Nectar, Sparks). This drives:
  - Customer Retention
  - Data Collection for insights
  - Brand Building
- **Digital Integration:** Companies consistently direct customers to their websites and apps for feedback, loyalty programs, and online ordering, indicating a strong digital presence and strategy.
- **Location & Diversity:** The presence of stores in high-traffic areas (stations, city centers) and a variety of meal options (including vegetarian wraps) caters to a broad customer base with diverse needs.

THE FRUITS OF OUR LABOUR:

The Tester Documentation Lead :

- Checked output
- verified folder and naming logic
- Assessed images and if text are usable and clear, wrote a report (in the form of a table)
- And as the Data Curator:
- Assesses date is usable for data extraction
- Reread data for accuracy / clarity.

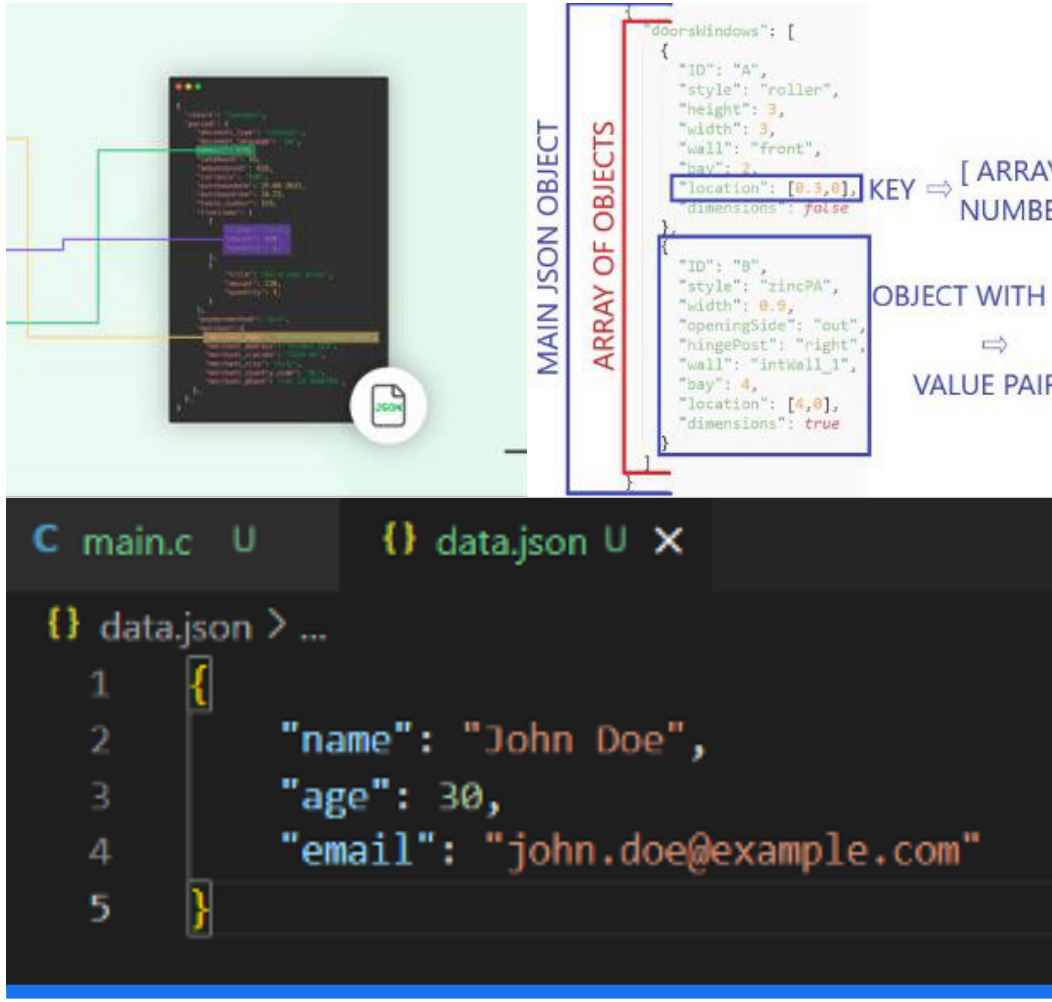
QA Review Table

Test Number	Test Description	Expected Outcome	Actual Outcome	Pass/Fail	Comments
1	Checked PDF-to-image conversion for Food 1–5	All PDFs should convert into PNG format	All PDFs successfully converted into PNG images	Pass	All PDF files were correctly converted into PNG format
2	Verified image readability and clarity	Images should be readable and clear	Images are clear and readable	Pass	All images are legible and of good quality
3	Checked if file names are logically organized	File names should be consistent and logically labeled	File names follow a logical and consistent structure	Pass	None
4	Confirmed each input PDF has a corresponding set of images	Each PDF should have a corresponding image folder with extracted pages	Each PDF maps correctly to its own image(s)	Pass	None
5	Verified data extracted matches content in original PDFs	Extracted data should be accurate and complete	The extracted text matches the content in the images	Pass	Each text output includes all receipt information with no missing

LOOK TO THE FUTURE:

Consider this...

- This step encourages the scalability of the program
- Having a structured data output – not just a text transcription of the receipts but one that responds to prompts to identify specific key information like “Date of Purchase”, "Name of Merchant" etc.
- The extracted information will be organized into a table that is read by the system itself, like a JSON file, like this ->
- Beneficial if the business has KPI’s for efficiency and accuracy – a collation of data in a structure or the recall of data, can mean that an employee or anyone who uses this programme can easily search for what they’re looking for and complete tasks as soon as possible.



SUMMARISE & CONCLUDE:

- Our AI-powered system is designed to automate the initial, time-consuming process of extracting information from scanned PDF receipts.
- Currently, the systems output is clear, readable text files that capture the exact content and visual layout of your original receipts. These text files provide a digital, searchable, and easily accessible version of your receipt data .
- It is essentially a smart assistant designed specifically to address the inefficiencies of manual data entry. This, results in the lessening of time-consuming tasks of manually extracting information from scanned PDFs receipts, that have been inputted into simple readable bits of information.
- We've demonstrated strong collaboration, problem-solving, and a deep understanding of AI applications in a business context.

References:

- accounts.google.com. (n.d.). *Sign in - Google Accounts*. [online] Available at: https://aistudio.google.com/app/apikey.
- Author), A.C. (PIL F. (n.d.). *Pillow: Python Imaging Library (Fork)*. [online] PyPI. Available at: https://pypi.org/project/Pillow/.
- Belval, E. (n.d.). *pdf2image: A wrapper around the pdftoppm and pdftocairo command line tools to convert PDF to a PIL Image list*. [online] PyPI. Available at: https://pypi.org/project/pdf2image/.
- googleapis (2024). *GitHub - googleapis/python-genai: Google Gen AI Python SDK provides an interface for developers to integrate Google's generative models into their Python applications*. [online] GitHub. Available at: https://github.com/googleapis/python-genai [Accessed 22 May 2025].
- oschwartz10612 (2024). *GitHub - oschwartz10612/poppler-windows: Download Poppler binaries packaged for Windows with dependencies*. [online] GitHub. Available at: https://github.com/oschwartz10612/poppler-windows [Accessed 22 May 2025].

- PyPI. (n.d.). *tqdm: Fast, Extensible Progress Meter*. [online] Available at: https://pypi.org/project/tqdm/.
- VickyNarotamo (2025). *GitHub - VickyNarotamo/East_Midlands_AI_Project: A student-led consultancy project using Python and Gemini AI to extract and summarize data from scanned PDFs for East Midlands Chamber. Developed as part of De Montfort University's Lead and Inspire programme.* [online] GitHub. Available at: https://github.com/VickyNarotamo/East_Midlands_AI_Project [Accessed 22 May 2025].
- "How to Resolve Manual Data Entry Errors with Automated Solutions." *Www.habiledata.com*, www.habiledata.com/blog/how-automated-solutions-resolve-data-entry-errors/.
- Martinez, Jorge. "DocuClipper." *DocuClipper*, 5 Mar. 2024, www.docuclipper.com/blog/data-entry-statistics/.
- PricewaterhouseCoopers. "Automating Data Extraction with AI." *PwC*, 24 June 2021, www.pwc.com/gx/en/issues/artificial-intelligence/publications/ai-automation-data-extraction.html.

That's all from Team Three. Thank you for listening to our presentation.

Q&A Time.

Here is our GitHub link and QR code for the project:

- https://github.com/VickyNarotamo/East_Midlands_AI_Project