Assessment: Build an AI Application for Arabic Invoice Processing

Objective:

Develop an application that integrates with **Google Drive API** to monitor a folder, retrieve Arabic invoices, and process them using an **Arabic OCR model**. The application should extract key information (e.g., invoice number, date, total amount, second product cost) from the invoice.

مشرق التجار ذ م م

123، مبنى ABC، شارع ديف، GHI

إدراج الشعار

TRN: AE12DE123456

فاتورة الضرائب

رقم الفاتورة:	١		تاريخ:	12/27/2021

معرف العميل: ش ب ط 001	نوع: Credit
اسم: أجهزة كمبيوتر الأسفلت	أيام: 15
عنوان: شارع ديف 1، ABC	تاريخ الاستحقاق: 1/11/2022
	TRN العميل: AE12DE123456

مبلغ	ضريبة %	القيمة للضريبة	الكمية	معدل	وصف المنتج	معرف المنتج	المسلسل
AED ٥٢٥٠٠	٥%	AED ٥٠٠٠٠	2	AED ٢٥٠٠٠	ديل إنسبيرون 1050	ب ب ط001	1
AED ٥٦٠٠٠	٠%	AED ٥٦٠٠٠	2	AED ٢٨٠٠٠	لينوفو I-5125	ب ب ط002	2
AED ٢٥٢٠	٥%	AED ٢٤٠٠	2	AED ١٢٠٠	لوجيتك ماوس اللاسلكية	ب ب ط004	3
AED ٥٠٠٠	٠%	AED ٥٠٠٠	2	AED ٢٥٠٠	لوجيتك لوحة المفاتيح	ب ب ط005	4
AED ٤٦٠٠	٠%	AED ٤٦٠٠	2	AED ٢٣٠٠	زيبروليكس لوحة المفاتيح	ب ب ط014	5
AED ١٠٥٠	٥%	AED ١٠٠٠	2	AED ٥٠٠	لوجيتك ماوس	ب ب ط003	6
AED ٥٦٠٠	٠%	AED ٥٦٠٠	2	AED ٢٨٠٠	زيبروليكس لوحة المفاتيح اللاسلكية	ب ب ط015	7
AED ١٦٨٠٠	٥%	AED ١٦٠٠٠	2	AED ٨٠٠٠	شاشة Dell 5684	ب ب ط011	8
AED ٣٧٨٠	٥%	AED ٣٦٠٠	2	AED ١٨٠٠	زيبروليكس ماوس	ب ب ط012	9
AED ٢١٠	٥%	AED ٢٠٠	2	AED ١٠٠	لوحة الماوس	ب ب ط009	10

AED ١٤٨٠٦٠	مجموع	AED ١٤٤٤٠٠	20

المبلغ بالكلمات	ملخص	
	المبلغ الخاضع للضريبة	AED ١٤٤٤٠٠
مائة وخمسة وأربعون ألف مائة واثنين وسبعين درهم	مبلغ ضريبة القيمة المضافة	AED ٣٦٦٠
	خصم 2%	AED ٢٨٨٨
ملاحظة: لن تقبل أي مطالبات بعد 24 ساعة من التسليم.	مبلغ الفاتورة	AED ١٤٥١٧٢

تلاحظ:

خ تم الشركة   الموقع المعتمد

شكرا لك على التعامل معنا

Assessment Details:

Step 1: Google Drive API Integration

- **Goal**: Integrate Google Drive API to monitor a specific folder for newly uploaded files.
- **Requirements**:
    - Authenticate using OAuth 2.0 for secure access.
    - Watch a designated Google Drive folder for changes (e.g., file upload events).
    - Automatically download the uploaded file when a change is detected.

Step 2: Process Arabic Invoice

- **Goal**: Extract and process key information from the downloaded invoice using an **Arabic OCR model**.
- **Requirements**:
    - Build or use an existing OCR model tailored for Arabic script (research and pick the best one with high accuracy). For extra credit, you can also use a lightweight custom OCR model built using **PyTorch**.
    - Process the invoice to extract:
        - **Invoice Number**
        - **Date**
        - **Second Product cost**
        - **Total Amount**
    - Return extracted information in JSON format.

Step 3: API Endpoints

- **Goal**: Provide a REST API to demonstrate the functionality.
- **Requirements**:
    - Create an API to:
        - Trigger Google Drive folder monitoring.
        - Manually upload and process an invoice as a fallback (e.g., for testing OCR without Google Drive).
        - Return extracted invoice details in JSON format.

Deliverables:

1. **Code**:
   - Modular, clean, and well-documented code uploaded to a GitHub repository.
   - Include setup instructions in a `README.md` file.
2. **Postman Collection**:
   - Include a Postman collection to demonstrate:
     - Triggering the Google Drive monitoring.
     - Uploading an invoice manually.
     - Retrieving the extracted invoice details.
3. **Demo**:
   - In the next meeting, you present the following:
     - Google Drive folder monitoring and file retrieval.
     - Arabic OCR model processing an invoice.
     - API endpoints in action.

Evaluation Criteria:

1. **Google Drive API Integration**:
   - Does the application successfully monitor the folder and retrieve new files?
2. **Arabic OCR Model**:
   - Does the OCR model correctly extract Arabic text from invoices?
   - Is the key information accurately extracted and structured in JSON format?
3. **API Functionality**:
   - Are the API endpoints functional, secure, and well-documented?
4. **Code Quality**:
   - Is the code clean, efficient, and maintainable?
5. **Postman Collection**:
   - Are all API functionalities well-documented and testable via Postman?

Bonus Points:

- Implement error handling for:
  - Invalid file formats.
  - Failed OCR processing.
  - Google Drive API errors.
- Provide clear instructions for deploying the application (e.g., in Docker).
- Use an advanced OCR model like **Tesseract fine-tuned for Arabic invoices** or a lightweight custom-trained PyTorch model.