Jour 2 : TP1
Lecture & Structuration d'un PDF

Parcours : Assistant vocal multimodal (.NET + Python + React + Riva)

Durée estimée : 1h – 1h30

Objectif pédagogique

Dans ce premier TP, vous allez apprendre à :

- Extraire le **contenu textuel d'un document PDF**.
- Extraire le **contenu textuel d'une image text PDF**.
- Nettoyer et structurer ce contenu dans un format exploitable (JSON).
- Préparer une base de **contexte textuel** que les prochains TPs pourront utiliser pour la synthèse ou la lecture vocale.

L'objectif n'est **pas** encore d'utiliser l'IA ou la voix, mais de produire un format **propre et structuré**, adapté à un assistant vocal.

Technologies et outils utilisés

Environnement principal

- **Langage :** C# (.NET 8 ou 9)
- **Type d'application :** Minimal API REST (backend léger)
- **IDE recommandé :** Visual Studio / Rider / VS Code
- **Contrôle de version :** Git + GitHub/GitLab (suivi de version et push du code)

Bibliothèques clés

Fonction	Outil / Package	Description
Lecture PDF	**PdfPig** (dotnet add package PdfPig --version 0.1.11)	Extraction de texte à partir de PDF textuels.
OCR (texte d'image)	**Tesseract** (dotnet add package Tesseract --version 5.2.0)	Reconnaissance optique de caractères sur les PDF scannés.
Conversion PDF → Image	**Docnet.Core** (dotnet add package Docnet.Core --version 2.6.0)	Conversion des pages PDF en images exploitables par Tesseract.
Manipulation d'images	**System.Drawing.Common** (dotnet add package System.Drawing.Common --version 8.0.0)	Traitement et rendu d'images avant OCR.
Documentation API	**Swashbuckle.AspNetCore** (dotnet add package Swashbuckle.AspNetCore --version 6.5.0)	Génération automatique de l'interface Swagger.

Livrables attendus

À la fin du TP, vous devez fournir :

Un service HTTP exposant un endpoint POST /pdf/parse :

- Entrée : fichier PDF (upload multipart/form-data).

Sortie : objet JSON suivant le schéma ci-dessous :

```
{
    "title": "NomDuFichier.pdf",
    "sections": [
      { "heading": null, "text": "..." },
      { "heading": "Section 1", "text": "..." }
    ],
    "meta": { "pages": 12 }
}
```

Connaissances préalables

Avant de commencer, vous devez savoir :

- Créer une **Minimal API .NET** avec dotnet new web.
- Gérer une **requête POST** avec upload de fichier (IFormFile).

- Comprendre la structure d’un PDF (texte, images, blocs).
- Manipuler du JSON et lire les résultats dans Postman.