



★ Mini-projet : Automatisation du traitement des RAW DATA en tableaux normalisés

1. 🎯 Objectif du projet

Développer un **script automatisé** (langage au choix) capable de transformer des **documents bruts Word (RAW DATA)** en un **document Word formaté (Processed DATA)** contenant des tableaux normalisés et exploitables. Ce projet a pour but d'évaluer :

- La **maîtrise technique** (parsing, automatisation, génération documentaire),
- La **compréhension métier** (essais CEM, marges, conformité),
- La **rigueur méthodologique** (structuration du code, documentation, traçabilité).

2. 📁 Données d'entrée et sortie

Entrée

Les candidats disposeront de **3 fichiers Word bruts (RAW DATA)** contenant :

- Des **paramètres de test** (nom du test, RBW, antenne, orientation DUT, step, pre-amp, etc.),
- Des **sections de mesures** (CISPR.AVG, Q-Peak, Peak, Limites associées),
- Des **tableaux chiffrés** (fréquence, mesures, limites, polarisation, corrections),
- Des **métadonnées** (sample ID, opérateur, date/heure, conclusion brute).

Sortie attendue

Un document Word **"Processed DATA"** qui présente :

- Un **cartouche en-tête** avec les paramètres de test,
- Des **tableaux normalisés et homogènes** :
 - Colonnes : Section | Frequency (MHz) | SR | Polarization | Correction (dB) | Mesure (dBµV/m) | Limite (dBµV/m) | Marge (dB) | Verdict
 - Mise en forme : tableau pro (bandeau coloré, lignes alternées, FAIL en rouge gras, PASS en vert),
- Une **synthèse de conformité** par section et par fichier.

En plus du Word, un **export CSV/Excel** est attendu avec la même structure.





3. Règles de traitement (clarification)

Détection des sections

- Reconnaître automatiquement :
 - CISPR.AVG / Lim.Avg
 - Q-Peak / Lim.Q-Peak
 - Peak / Lim.Peak
- Les variations d'intitulés (majuscule/minuscule, espaces, abréviations) doivent être gérées.

Extraction des données

- Conserver toutes les lignes **chiffrées** des tableaux (ne pas perdre d'information).
- Nettoyer et homogénéiser :
 - dB μ V/m \rightarrow dB μ V/m
 - , décimale \rightarrow .
 - Normalisation des colonnes (espaces, majuscules).

Logique métier (conformité)

- **Marge (dB) = Limite – Mesure**
- **Verdict ligne :**
 - PASS si marge ≥ 0.00 dB,
 - FAIL si marge < 0.00 dB.
- **Verdict section :** PASS si toutes les lignes PASS, sinon FAIL.
- **Verdict global (document) :** PASS si toutes les sections PASS.
- En cas d'écart entre marge calculée et marge fournie (> 0.05 dB) \rightarrow recalcul + log.

Normalisation des unités et valeurs

- **Fréquence :** 5 décimales < 10 MHz, sinon 3 décimales.
- **dB :** 2 décimales.
- **Polarization :** conserver Vertical / Horizontal.
- **RBW :** 9 kHz, 120 kHz, 1 MHz (doivent être correctement détectés et affichés).





4. 📁 Structure du livrable attendu

Le candidat devra livrer une **archive complète** :

```
MiniProjet_CCC <NomCandidat>/
├── README.md (instructions d'installation & exécution, choix techniques)
├── /src/ (code source bien structuré)
│   ├── main.<lang>
│   ├── parser.<lang>
│   ├── rules.<lang>
│   ├── writer.<lang>
│   └── utils.<lang>
├── /tests/ (fichiers de test unitaires + fixtures RAW fournis)
├── /out/ (résultats générés)
│   ├── Processed_RAW01.docx
│   ├── Processed_RAW01.csv
│   ├── Processed_RAW02.docx
│   └── Processed_RAW02.csv
├── logs/
└── /config/ (regex, mapping colonnes, options)
```

Chaque fichier Word généré doit contenir :

1. **Cartouche d'en-tête** (Sample ID, RBW, Antenne, Orientation DUT, etc.),
2. **Tableaux formatés** (par section de mesure),
3. **Synthèse finale** avec le verdict global (OK/NOK).

5. 🛠 Langages autorisés

Le projet peut être réalisé dans l'un des environnements suivants :

- **Python** (recommandé) : python-docx, pandas, regex, docx2txt,
- **Node.js** : mammoth, docx, xlsx,
- **C# / Java** : OpenXML SDK / Apache POI,
- **VBA** (Word ou Excel) : **macro intégrée pour automatiser directement.**

6. ✔ Critères d'évaluation

- Exactitude du parsing (30%) → toutes les lignes doivent être extraites, données fidèles, encodage correct.
- Application de la logique métier (25%) → marges et verdicts corrects.
- Qualité du rendu (20%) → Word bien formaté, tableaux propres, lisibles et professionnels.
- Robustesse du script (15%) → gère encodages, dates invalides, sections manquantes.
- Qualité du code et de la doc (10%) → structure claire, commentaires, README complet.





7. ☹ Anti-triche & encadrement

- Le code source doit être **original** (pas de simple copie en ligne).
- Chaque résultat doit inclure une **signature automatique** en pied de page Word :
- <NomCandidat> | <Date d'exécution> | <Hash du fichier RAW>
- Les candidats doivent fournir leurs **logs d'exécution** pour prouver l'authenticité.

