PROJET STATION TV

SPÉCIFICATIONS FONCTIONNELLES ET TECHNIQUES

SUIVI DES MODIFICATIONS DU DOCUMENT			
Version	Date	Validation	Commentaire
1.0	08/10/2025	Frédéric Chauvin	Rédaction initiale

REDACTEUR : Dorian BRISSON	VALIDATEUR : Frédéric CHAUVIN
CLIENT : Polytech Tours	ENCADRANT : Mathieu DELALANDRE

OBJET DU DOCUMENT

Il complète le Cahier des Charges et le Plan de Développement du projet, en apportant une description approfondie de la mise en œuvre concrète du système de transcription audio basé sur Whisper (OpenAI), utilisé au sein de la plateforme Station TV du Laboratoire LIFAT.

Ce document a pour but de :

- Décrire les fonctions logicielles principales à développer ou adapter (pipeline Whisper, scripts de monitoring, supervision QoS),
- Définir l'architecture logicielle et matérielle de la station Dell 5820 utilisée pour le projet,
- Préciser les caractéristiques techniques des modules logiciels (paramètres, flux, dépendances),
- Formaliser les procédures de test et validation,
- Et identifier les contraintes, limites et perspectives d'évolution associées au système.

Les spécifications présentées ici serviront de référence technique unique tout au long du projet, garantissant la cohérence entre les différentes étapes :

- Configuration matérielle,
- Optimisation du pipeline Whisper,
- Exécution des tests à grande échelle,
- Et validation finale de la transcription audio haute performance.

Ce document sera également utilisé comme support d'évaluation et de maintenance pour les futures évolutions du projet Station TV, notamment dans le cadre des travaux de recherche du LIFAT sur la reconnaissance multimodale et la transcription massive de données audiovisuelles.

CONTEXTE DU PROJET

Le projet Station TV – Transcription audio haute performance s'inscrit dans la continuité du programme de recherche Station TV, développé depuis 2019 par le Laboratoire d'Informatique Fondamentale et Appliquée de Tours (LIFAT), au sein de l'équipe RFAI (Reconnaissance des Formes et Analyse d'Images).

L'objectif global du projet Station TV est de concevoir une infrastructure de calcul parallèle capable de capturer, traiter et analyser automatiquement les flux télévisuels issus de la TNT française.

Dans ce cadre, la transcription audio joue un rôle essentiel : elle constitue la porte d'entrée textuelle du contenu audiovisuel, permettant d'alimenter les modules d'analyse linguistique, de recherche d'entités et de sémantique TV.

Une première solution de transcription a été développée en 2024–2025 par T. Bourdeau dans le cadre du même projet.
Cette version repose sur Whisper, le modèle open source d'OpenAI, et permet la transcription en batch de flux audio à partir des chaînes TNT enregistrées.
Toutefois, des limitations techniques ont été observées lors du traitement à haute qualité :

- Saturation mémoire au-delà du modèle *medium*,
- Temps de traitement trop long sur les séquences de plusieurs heures,
- Instabilité lors de l'exécution parallèle (multi-thread CPU),
- Manque de supervision des performances (QoS).

Le présent projet a donc pour but de faire évoluer l'architecture matérielle et logicielle de la Station TV afin de lever ces limitations.

Concrètement, les objectifs principaux sont :

1. Mettre à niveau la station de calcul Dell Precision 5820, en augmentant la mémoire vive à 256 Go DDR4 ECC, et en assurant la compatibilité BIOS et la stabilité du système.
2. Optimiser le pipeline Whisper afin de rendre la transcription plus rapide, stable et scalable, sur des modèles de grande taille (*medium*, *large*).
3. Mettre en place un système de métriques QoS permettant de suivre automatiquement les performances du pipeline (temps de traitement, consommation CPU/RAM, débit, précision linguistique).

Le projet est conduit par D. Brisson, étudiant de 5e année ISIE, sous la direction de M. Mathieu Delalandre.
La DSI de Polytech Tours assure le support matériel (installation et configuration de la RAM) et la maintenance de la station Dell.

ARCHITECTURE GÉNÉRALE

Vue d'ensemble

Le système développé dans le cadre du projet Station TV – Transcription audio haute performance repose sur une architecture modulaire combinant plusieurs composants matériels et logiciels, interconnectés au sein d'un pipeline de traitement automatique de flux audio.

L'objectif est de permettre la transcription à grande échelle de données audio issues de la TNT, tout en garantissant des performances mesurables, une supervision continue et une scalabilité adaptée aux besoins du laboratoire LIFAT.

Le système se compose de quatre modules principaux, articulés autour de la station de calcul Dell Precision 5820 :

1. Module de Capture et Prétraitement Audio
    - Gère la récupération et la segmentation des flux TNT en fichiers audio exploitables.
    - Prépare les données d'entrée pour la transcription (formats, durée, encodage).
2. Module de Transcription (Pipeline Whisper)
    - Réalise la conversion audio → texte via les modèles Whisper (*tiny* → *large*).
    - Implémente les optimisations CPU/mémoire et le traitement batch/multi-process.
3. Module de Supervision et Mesure de Performance (QoS)
    - Surveille la charge CPU, la consommation mémoire et le temps de traitement.
    - Génère automatiquement les rapports de performance et de stabilité.
4. Module d'Export et d'Intégration Station TV
    - Convertit les transcriptions en formats exploitables (TXT, CSV, JSON).
    - Intègre les résultats dans les bases de données et le pipeline multimodal de la Station TV.

L'ensemble fonctionne sur un environnement multi-thread CPU optimisé pour la robustesse et la montée en charge, sans recours à l'accélération GPU (contraintes matérielles et stabilité privilégiée).

Circulation des données

1. **Entrées :**
   - Fichiers audios au format MP3, 256 kbps, 48 kHz, issus de la capture TNT.
   - Métadonnées associées (chaîne, date, heure, émission).
2. **Traitement :**
   - Le module Pipeline Whisper lit les fichiers audios, segmente les flux et exécute le modèle sélectionné.
   - Les processus sont répartis sur les 36 threads CPU pour maximiser le débit sans saturer la RAM.
   - Les métriques QoS sont collectées en temps réel.
3. **Sorties :**
   - Fichiers texte (TXT, CSV, JSON) contenant la transcription alignée.
   - Rapports QoS (CPU, RAM, temps, débit, WER).
   - Logs techniques (succès/échec, erreurs système, durée).

Caractéristiques de l'architecture

Élément	Spécification
Machine hôte	Dell Precision 5820 – Xeon W-2295 (18 cœurs / 36 threads), 256 Go DDR4 ECC
OS	Windows Server 2022 / Ubuntu 22.04
Langage principal	Python 3.10
Bibliothèques principales	Whisper (OpenAI), Torch, NumPy, Pandas, FFmpeg, psutil
Type de traitement	CPU multi-process (sans GPU)
Durée de session typique	1 semaine d'exécution continue

Objectif de conception

Cette architecture a été pensée pour répondre à trois priorités :

1. **Performance :** permettre des traitements rapides et simultanés sur de grands volumes audio sans saturation mémoire.
2. **Robustesse :** assurer une exécution stable, y compris sur des sessions longues.
3. **Observabilité :** collecter en continu les indicateurs de qualité de service pour ajuster les paramètres et optimiser les performances globales.

SPÉCIFICATIONS FONCTIONNELLES

Le système est composé de quatre modules fonctionnels principaux, interconnectés au sein du pipeline Station TV :

1. Module Capture & Prétraitement Audio
2. Module de Transcription Whisper (STT)
3. Module de Supervision & QoS
4. Module d'Export et d'Intégration Station TV

Chaque module est décrit ci-dessous selon ses entrées, sorties, algorithmes, paramètres clés et contraintes de performance.

Module Capture & Prétraitement Audio

Objectif :
Préparer les fichiers audios provenant des flux TNT pour les rendre exploitables par le pipeline Whisper.
Ce module constitue la première étape du traitement, en garantissant la qualité du signal, le formatage et la segmentation temporelle des données.

Élément	Description
Entrées	Fichiers MP3 (256 kbps / 48 kHz) issus de la capture TNT, segmentés par émission et chaîne.
Sorties	Fichiers WAV normalisés, durée fixe (5–20 min), avec métadonnées horodatées.
Paramètres	- Durée segment : 10 min - Fréquence d'échantillonnage : 48 kHz - Format : PCM16 (mono ou stéréo)
Fonctions clés	- Conversion automatique par lot - Nettoyage des silences (seuil dB) - Vérification de la durée et de l'intégrité audio
Critères de performance	- 100 % des fichiers convertis - Erreur de conversion < 0,1 % - Durée moyenne traitement < 10 s / fichier

Module de Transcription Whisper (STT)

Objectif :
Convertir les fichiers audios en texte brut à l'aide du modèle Whisper (OpenAI), avec un compromis entre rapidité, précision et charge mémoire.

Élément	Description
Entrées	Fichiers WAV normalisés produits par le module de capture.
Sorties	Fichiers texte (.txt / .csv) contenant la transcription alignée par segment.
Algorithme / Modèle	Modèle Whisper (versions *tiny, base, small, medium, large*).
Paramètres configurables	- Langue : FR - Taille du modèle - Taille des lots (*batch size*) - Threads simultanés (≤ 36) - Seuil de confiance (0.6–0.9)
Fonctions clés	- Détection automatique de la langue - Segmentation dynamique - Nettoyage des balises et caractères spéciaux - Gestion du multi-thread CPU
Critères de performance	- Débit ≥ 5× temps réel pour modèle *small* - Débit ≥ 1× temps réel pour modèle *medium* - Word Error Rate (WER) ≤ 10 % sur segments clairs
Contraintes	- Exécution CPU uniquement (pas de GPU) - RAM disponible : 256 Go (RAM ECC) - Environnement Python isolé (virtualenv)
Scripts principaux	BasicTestWhisper.py (test unitaire), RunBatchWhisper.py (batch multi-process)

Module de Supervision & Qualité de Service (QoS)

Objectif :
Assurer le suivi en temps réel de l'utilisation CPU, de la mémoire, du débit de transcription et de la qualité linguistique (WER), afin d'évaluer les performances globales du système.

Élément	Description
Entrées	Logs Whisper générés à chaque exécution (durée, taille, modèle, threads).
Sorties	Fichiers CSV + graphiques (.png) contenant les mesures CPU/RAM, temps de traitement et débit.
Algorithme / Méthode	Surveillance via psutil, calculs statistiques via Pandas, visualisation via Matplotlib.
Paramètres	- Fréquence d'échantillonnage CPU/RAM : 2 s - Fenêtre de calcul du débit : 1 min - Calcul WER sur 5 échantillons / run
Fonctions clés	- Calcul automatique du throughput (heures traitées / heure réelle) - Graphiques CPU/RAM sur la durée totale d'exécution - Export QoS automatique après chaque run
Critères de performance	- Suivi 100 % des sessions - Écart max CPU/RAM ≤ 5 % entre runs consécutifs - Logs exploitables sous Excel / Python
Scripts principaux	ComputeQoS.py, QoSMonitor.py
Contraintes	- Compatible uniquement Windows Server / Ubuntu 22.04 - Nécessite droits administrateur (supervision mémoire globale)

Module d'Export et d'Intégration Station TV

Objectif :
Assurer l'export des transcriptions et leur intégration dans l'écosystème de la Station TV, pour exploitation par les modules de recherche et d'analyse.

Élément	Description
Entrées	Fichiers texte (.txt / .csv) produits par le module Whisper.
Sorties	Données formatées (.json / .csv) et indexées dans la base Station TV.
Algorithme / Méthode	Conversion structurée + normalisation des champs (chaîne, date, heure, transcription).
Paramètres	- Encodage UTF-8 - Format ISO-8601 pour les dates - Indexation par ID d'émission (collection)
Fonctions clés	- Fusion automatique des fichiers de transcription - Nettoyage et harmonisation des textes - Export vers le répertoire partagé du LIFAT
Critères de performance	- 100 % des fichiers exportés sans erreur - Intégration < 5 s / fichier - Aucune perte de donnée textuelle
Contraintes	- Conformité avec la base Station TV (structure et nommage) - Export automatique après chaque batch

SPÉCIFICATIONS TECHNIQUES

Le système de transcription à grande échelle repose sur une infrastructure matérielle hautes performances, un ensemble d'outils open source spécialisés pour le traitement audio et la supervision, et une configuration logicielle stable adaptée aux environnements CPU multi-thread.

Les spécifications suivantes définissent les caractéristiques minimales requises pour le fonctionnement, les tests et la maintenance du pipeline Whisper dans le cadre du projet Station TV.

Infrastructure matérielle

Domaine	Spécifications
Machine principale	Dell Precision 5820 Tower
Processeur (CPU)	Intel Xeon W-2295 – 18 cœurs / 36 threads – 3,0 GHz
Mémoire vive (RAM)	256 Go DDR4 ECC – 4 × 64 Go Kingston KTD-PE432/64G
Stockage	SSD 512 Go (système) + RAID 38 To (données TNT) + NAS 190 To
Carte graphique (non utilisée)	NVIDIA Quadro RTX 4000 (GPU désactivé)
Système d'exploitation	Windows Server 2022 / Ubuntu 22.04 LTS (dual boot)
Connectivité	Réseau 10 Gb/s – accès LIFAT et stockage distant
Onduleur / sauvegarde	Système UPS 1500VA – arrêt sécurisé en cas de coupure
Support matériel	DSI Polytech Tours – supervision de la station Dell

Objectif de performance matérielle : permettre l'exécution stable du modèle Whisper large sur 36 threads CPU avec une occupation RAM maximale < 95 %.

Environnement logiciel

Composant	Version / Détail	Rôle
Langage principal	Python 3.10	Langage de développement principal
Framework STT	Whisper (OpenAI) – version 1.5	Transcription audio → texte
Bibliothèque IA	Torch (PyTorch) 2.2.0 (CPU build)	Support mathématique et tensoriel
Librairies audio	FFmpeg 6.1, Librosa 0.10	Conversion et extraction audio

Composant	Version / Détail	Rôle
Librairies data	NumPy 1.26, Pandas 2.2	Calculs et analyses statistiques
Librairies graphiques	Matplotlib 3.9, Seaborn 0.13	Visualisation QoS
Supervision système	psutil 6.0	Monitoring CPU/RAM
Multi-processing	Python stdlib multiprocessing/threading	Exécution parallèle
Versionning	Git	Gestion du code et suivi des versions
IDE principal	Visual Studio Code 1.89	Environnement de développement
Gestion de paquets	pip + virtualenv	Isolation des environnements Python
Scripts principaux	BasicTestWhisper.py, RunBatchWhisper.py, ComputeQoS.py	Fonctions principales du pipeline

Paramètres d'exécution du pipeline Whisper

Élément	Spécification
Langue principale	Français (FR)
Formats d'entrée audio	MP3 / WAV / PCM16
Durée des fichiers traités	5 à 20 minutes par segment
Batch processing	Taille de lot : 2–6 fichiers simultanés
Threading CPU	12 à 36 threads (configurable)
Mémoire allouée par processus	24 à 64 Go
Sorties	.txt, .csv, .json (encodage UTF-8)
Fréquence d'échantillonnage	48 kHz
Nombre de runs simultanés	≤ 3 (testés en conditions sûres)
Durée d'une campagne complète	7 à 14 jours

Structure logicielle et scripts

La logique du projet repose sur un ensemble de scripts Python interconnectés, organisés par fonction :

Script	Fonction principale	Entrées	Sorties
BasicTestWhisper.py	Test unitaire des modèles Whisper	Fichiers WAV courts	Fichier texte brut
RunBatchWhisper.py	Transcription batch multi-process	Répertoires audio TNT	Dossiers TXT / CSV
ComputeQoS.py	Calcul des métriques QoS	Logs CPU/RAM	Tableaux CSV + graphiques
BatchManager.py	Coordination des runs Whisper	Liste des lots / modèles	Fichiers d'exécution
ExportResults.py	Conversion / Intégration Station TV	Dossiers TXT	JSON / CSV indexés
Monitor.py	Supervision CPU/RAM temps réel	Processus actifs	Alertes et logs techniques

Tous les scripts sont versionnés, documentés (docstring + fichier README.md) et exécutables en ligne de commande (CLI) ou via tâche planifiée.

Contraintes techniques

- **Compatibilité OS :** Windows Server 2022 (production) et Ubuntu 22.04 (test).
- **Exécution CPU-only :** pas de GPU utilisé pour garantir la stabilité et la reproductibilité.
- **RAM :** chaque processus Whisper *medium/large* consomme entre 40 et 90 Go ; exécution limitée à 2–3 processus simultanés.
- **Sauvegardes automatiques :** logs et résultats exportés quotidiennement sur le NAS LIFAT.
- **Sécurité :** accès restreint à la station (compte administrateur DSI uniquement).

Objectifs de performance mesurables

Indicateur	Objectif cible	Méthode de mesure
Débit moyen	≥ 5× temps réel (*small*) / ≥ 1× (*medium*)	Calcul throughput (QoS script)
WER (Word Error Rate)	≤ 10 % sur segments clairs	Échantillonnage sur 1h audio
Utilisation RAM moyenne	< 90 % pendant exécution	Script psutil / logs QoS
Utilisation CPU	> 85 % (36 threads actifs)	Supervision psutil
Stabilité session longue (>10h)	Aucune erreur / fuite mémoire	Test endurance
Taux de réussite transcription	100 % des fichiers traités	Comptage des sorties valides

TESTS ET VALIDATION

Les tests ont pour objectif de garantir la conformité du système par rapport aux exigences définies dans le Cahier des Charges et les spécifications fonctionnelles. Ils visent à valider à la fois les performances techniques (vitesse, stabilité, mémoire) et la qualité de transcription du système Whisper intégré dans la Station TV.

Chaque test correspond à une étape spécifique du pipeline : capture, transcription, supervision, export.
Les résultats serviront à la validation finale du projet PRI, ainsi qu'à la qualification du pipeline Station TV pour une utilisation à grande échelle.

Tableau de validation des tests

Type de test	Objectif	Critère de validation	Outil / Script associé
Test de performance – Whisper medium/large	Valider la stabilité et le débit sur modèles lourds	Débit ≥ 1× temps réel, RAM ≤ 240 Go, aucune erreur mémoire	RunBatchWhisper.py, psutil
Test QoS – Monitoring CPU/RAM	Mesurer la charge CPU et RAM pendant la transcription	Graphiques cohérents, écart CPU < 5 % entre runs	ComputeQoS.py
Test d'endurance – session longue	Vérifier la stabilité sur exécution prolongée	Aucun crash, débit stable, logs complets	RunBatchWhisper.py, Monitor.py
Test d'intégration – Export Station TV	Vérifier la structure et le format des fichiers exportés	Données UTF-8, formats JSON/CSV valides, structure conforme	ExportResults.py
Test de fiabilité – sauvegardes automatiques	Vérifier la présence des backups et des logs journaliers	Tous les logs présents, datés, et complets	Cron job / tâche planifiée Windows
Test qualité linguistique (WER)	Évaluer la précision de la transcription audio-texte	WER ≤ 10 % sur segments parlés clairs	WERMetric.py, référence manuelle
Test global – pipeline complet	Vérifier le fonctionnement global du système (entrée → sortie)	100 % des fichiers traités sans erreur, logs complets	Tous les scripts intégrés

Critères de réussite

Les critères de validation pour considérer le système comme opérationnel et conforme sont :

Domaine	Critère de réussite
Stabilité	0 crash sur longue exécution, logs complets sur 100 % des runs
Mémoire	Utilisation RAM < 95 % pendant les tests lourds
Qualité transcription (WER)	≤ 10 % sur segments clairs, ≤ 20 % sur audio complexe
Supervision QoS	Fichiers CSV/PNG cohérents et exportés à chaque session
Export Station TV	Fichiers JSON/CSV valides et exploitables sans erreur
Reproductibilité	Résultats similaires sur 3 exécutions successives

Outils et environnement de test

Outil	Rôle	Version
Python 3.10	Langage principal de développement	3.10.12
FFmpeg	Conversion et découpe audio	6.1
Whisper (OpenAI)	Moteur de transcription	1.5
Torch (PyTorch)	Calcul tensoriel CPU	2.2.0
psutil / Pandas / Matplotlib	Supervision et reporting QoS	6.0 / 2.2 / 3.9
Excel / Google Sheets	Analyse et synthèse des résultats	-
Dell Precision 5820 (256 Go ECC)	Plateforme principale de test	BIOS à jour

Conclusion

Ce document de Spécifications Fonctionnelles et Techniques synthétise l'ensemble des choix matériels, logiciels et organisationnels du projet.
Il constitue la référence technique unique pour le développement, la validation et la maintenance du système Station TV – Transcription audio haute performance.

Les modules décrits ici seront implémentés et validés selon le plan de tests, avant la livraison du rapport final PRI (janvier 2026).
Les résultats permettront de confirmer la faisabilité d'une transcription audio à grande échelle sur architecture CPU.