

Rapport Technique

Date : 18 septembre 2024

Auteurs : Christophe Chervet, Patrick Agù

Service : Informatique

Table des Matières

1. [Introduction](#)
 2. [Objectifs du Projet](#)
 3. [Analyse des Technologies Disponibles](#)
 - 3.1 [CMU Sphinx \(PocketSphinx\)](#)
 - 3.2 [Kaldi](#)
 - 3.3 [Mozilla DeepSpeech](#)
 - 3.4 [Vosk](#)
 4. [Choix de la Solution](#)
 5. [Exigences Techniques](#)
 - 5.1 [Matériel](#)
 - 5.2 [Logiciel](#)
 6. [Procédure d'Installation et de Configuration](#)
 - 6.1 [Installation de Python](#)
 - 6.2 [Installation des Bibliothèques Nécessaires](#)
 - 6.3 [Téléchargement du Modèle de Langue Français](#)
 7. [Développement du Système de Dictée Vocale avec Interface Graphique](#)
 - 7.1 [Choix du Framework d'Interface Graphique](#)
 - 7.2 [Structure du Programme](#)
 - 7.3 [Code Source](#)
 - 7.4 [Explication du Code](#)
 8. [Déploiement sur les Postes de Travail](#)
 - 8.1 [Création d'un Exécutable](#)
 - 8.2 [Distribution et Installation](#)
 9. [Guide d'Utilisation](#)
 10. [Maintenance et Support](#)
 - 10.1 [Mise à Jour du Modèle](#)
 - 10.2 [Résolution des Problèmes Courants](#)
 11. [Sécurité et Confidentialité](#)
 12. [Conclusion](#)
 13. [Annexes](#)
-

1. Introduction

Afin d'améliorer l'accessibilité et l'efficacité au sein de notre entreprise, ce rapport présente le développement d'un système de dictée vocale open source fonctionnant hors ligne, doté d'une **interface graphique** pour rendre l'utilisation plus agréable et facile. Ce système permettra aux employés, y compris ceux ayant des limitations physiques temporaires, de saisir du texte par la voix tout en garantissant la confidentialité des données.

2. Objectifs du Projet

- **Confidentialité** : Garantir que les données vocales ne quittent pas les postes de travail.
- **Accessibilité** : Faciliter la saisie de texte pour tous les employés.
- **Facilité d'Utilisation** : Ajouter une interface graphique intuitive.
- **Facilité de Déploiement** : Assurer une installation simple sur différents postes de travail.
- **Coût** : Utiliser des technologies open source pour éviter les coûts de licence.

3. Analyse des Technologies Disponibles

3.1 CMU Sphinx (PocketSphinx)

- **Avantages** : Léger, fonctionne hors ligne.
- **Inconvénients** : Précision inférieure par rapport aux solutions modernes.

3.2 Kaldi

- **Avantages** : Très puissant et flexible.
- **Inconvénients** : Complexité élevée, difficile à déployer.

3.3 Mozilla DeepSpeech

- **Avantages** : Basé sur l'apprentissage profond, bonne précision.
- **Inconvénients** : Exige des ressources matérielles importantes.

3.4 Vosk

- **Avantages** :
 - Bonne précision.
 - Faible utilisation des ressources.
 - Supporte plusieurs langues dont le français.
 - Facile à intégrer et à déployer.
- **Inconvénients** : Communauté moins large que certaines alternatives.

4. Choix de la Solution

Vosk a été sélectionné pour les raisons suivantes :

- **Fonctionnement hors ligne** : Respecte les exigences de confidentialité.
- **Facilité d'intégration** : API disponibles pour plusieurs langages.
- **Performances** : Fonctionne sur des machines avec des ressources limitées.
- **Prise en charge du français** : Modèles pré-entraînés disponibles.
- **Possibilité d'Interface Graphique** : Compatible avec les frameworks GUI en Python.

5. Exigences Techniques

5.1 Matériel

- **Processeur** : Dual-core 2 GHz ou supérieur.
- **Mémoire RAM** : Minimum 2 Go.
- **Espace Disque** : 500 Mo pour l'installation.
- **Microphone** : Intégré ou externe, de bonne qualité.

5.2 Logiciel

- **Système d'Exploitation** :
 - Windows 10 ou supérieur.
 - Linux (Ubuntu 18.04+).
 - macOS.
- **Python** : Version 3.6 ou supérieure.
- **Bibliothèques Python** :
 - vosk
 - sounddevice ou pyaudio
 - tkinter (inclus avec Python pour les interfaces graphiques)
- **Framework GUI** :
 - Utilisation de Tkinter pour l'interface graphique.

6. Procédure d'Installation et de Configuration

6.1 Installation de Python

Windows :

1. Télécharger l'installateur depuis python.org.
2. Lancer l'installateur et cocher "Add Python to PATH".

6.2 Installation des Bibliothèques Nécessaires

```
pip install vosk sounddevice
```

6.3 Téléchargement du Modèle de Langue Français

1. Accéder à Vosk Models.
2. Télécharger `vosk-model-small-fr-0.22`.
3. Extraire le fichier dans un répertoire dédié, par exemple `C:\vosk_models\fr`.

7. Développement du Système de Dictée Vocale avec Interface Graphique

7.1 Choix du Framework d'Interface Graphique

Nous avons choisi **Tkinter** pour les raisons suivantes :

- **Intégré** : Inclus par défaut avec Python.
- **Simplicité** : Facile à utiliser pour des interfaces simples.
- **Compatibilité** : Fonctionne sur Windows, Linux et macOS.

7.2 Structure du Programme

1. **Chargement du Modèle** : Initialiser le modèle de langue française.
2. **Création de l'Interface Graphique** : Construire la fenêtre principale avec les boutons nécessaires.
3. **Capture Audio** : Utiliser le microphone pour capturer l'audio en temps réel lors de l'appui sur le bouton "Démarrer".
4. **Traitement** : Transcrire l'audio en texte à l'aide de Vosk.
5. **Affichage** : Afficher le texte transcrit dans une zone de texte de l'interface.
6. **Arrêt de la Capture** : Permettre à l'utilisateur d'arrêter la dictée.

7.3 Code Source

Index.py

7.4 Explication du Code

- **Importation des Modules** : Ajout de `tkinter` pour l'interface graphique et `threading` pour gérer les threads.
- **Classe VoiceDictationApp** :
 - **init** : Initialisation de l'interface graphique avec une zone de texte et des boutons "Démarrer" et "Arrêter".
 - **audio_callback** : Capture les données audio et les ajoute à la file d'attente.
 - **start_listening** : Active la capture audio et lance le thread de reconnaissance.
 - **stop_listening** : Désactive la capture audio et arrête le thread.
 - **listen_and_recognize** : Traite l'audio en temps réel et affiche le texte dans la zone de texte.
- **Gestion des Threads** : Utilisation de `threading.Thread` pour ne pas bloquer l'interface graphique lors de la capture audio.
- **Interface Graphique** : Utilisation de `ScrolledText` pour permettre le défilement du texte transcrit.
- **Gestion des Exceptions** : Affichage d'une boîte de dialogue en cas d'erreur.

8. Déploiement sur les Postes de Travail

8.1 Création d'un Exécutable

Pour faciliter le déploiement, le script Python peut être converti en exécutable.

Utilisation de PyInstaller :

Installer PyInstaller :

```
pip install pyinstaller
```

Générer l'exécutable :

```
pyinstaller --onefile --add-data "C:/vosk_models/fr;vosk_models/fr" dictée_vocale_gui.py
```

- `--onefile` : Crée un seul fichier exécutable.
- `--add-data` : Inclut le modèle de langue dans l'exécutable.
- `dictée_vocale_gui.py` : Nom du script Python avec interface graphique.

8.2 Distribution et Installation

- **Distribution :**
 - Fournir l'exécutable aux employés via le réseau interne ou un support de stockage.
- **Installation :**
 - Copier l'exécutable sur le poste de travail.
 - S'assurer que les pilotes du microphone sont à jour.

9. Guide d'Utilisation

1. **Lancement :**
 - Double-cliquer sur l'exécutable `dictée_vocale_gui.exe`.
 - Une fenêtre s'ouvre avec une zone de texte et deux boutons : "Démarrer la dictée" et "Arrêter la dictée".
2. **Utilisation :**
 - Cliquer sur "Démarrer la dictée" pour commencer.
 - Parler clairement en direction du microphone.
 - Le texte transcrit apparaît dans la zone de texte.
 - Cliquer sur "Arrêter la dictée" pour terminer.
3. **Fonctionnalités Supplémentaires :**
 - Le texte peut être sélectionné, copié ou sauvegardé à partir de la zone de texte.
 - En cas d'erreur, un message s'affiche pour guider l'utilisateur.

10. Maintenance et Support

10.1 Mise à Jour du Modèle

- **Téléchargement :** Obtenir la dernière version du modèle depuis Vosk Models.
- **Remplacement :** Mettre à jour le répertoire du modèle sur les postes de travail.

10.2 Résolution des Problèmes Courants

- **Aucune Transcription :**
 - Vérifier que le microphone est correctement connecté.
 - Tester le microphone avec une autre application.
- **Transcription Incorrecte :**
 - Réduire le bruit ambiant.
 - Vérifier la qualité du microphone.
- **L'application ne se lance pas :**
 - S'assurer que le système répond aux exigences minimales.
 - Vérifier que toutes les dépendances sont incluses dans l'exécutable.
- **Erreurs d'Audio :**
 - Des messages d'erreur audio peuvent s'afficher en cas de problème avec le périphérique audio.

11. Sécurité et Confidentialité

- **Traitement Local** : Toutes les données sont traitées localement, aucune donnée n'est envoyée sur Internet.
- **Absence de Stockage Permanent** : Les données audio ne sont pas enregistrées sur le disque.
- **Code Open Source** : Permet la vérification et l'audit du code pour s'assurer de l'absence de failles de sécurité.
- **Interface Graphique Sécurisée** : Aucune fonctionnalité n'est incluse pour exporter ou partager les données transcrites en dehors de l'application.

12. Conclusion

Le système de dictée vocale développé avec une interface graphique améliore significativement l'expérience utilisateur. Il répond pleinement aux besoins de l'entreprise en offrant une solution sécurisée, accessible et facile à déployer. Cette application permettra d'améliorer la productivité et l'inclusion au sein de l'entreprise, tout en respectant les normes de confidentialité.

13. Annexes

- **Liens Utiles** :
 - Site Officiel de Vosk
 - [Documentation de Vosk sur GitHub](#)
 - Téléchargement des Modèles de Langue
 - **Contact Support** :
 - **Chervet Christophe** : CHRISTOPHE.CHERVET@assurance-maladie.fr
 - **Agù Patrick** : PATRICK.AGU@assurance-maladie.fr
-