

# **DOCUMENTATION TECHNIQUE DE L'APPLICATION VOCAL WEATHER**

## 1 - Introduction

- L'application Vocale Weather permet à un utilisateur de demander la météo à l'aide de commandes vocales
- Elle utilise FastAPI pour le backend, Open-Meteo API pour les données météo, et Azure Cognitive Services pour la reconnaissance vocale

## 2 - Pré-requis et Installation

### Prérequis

- Python 3.9+
- Virtualenv (recommandé)
- Accès à l'API Open-Meteo
- Accès à l'API Azure Speech-to-Text

### Installation

- Cloner le projet
- Créer un environnement virtuel
- Installer les dépendances Python
- Configurer les variables d'environnement
- Lancer l'API FastAPI

### 3 - Structure du Projet

-- main.py	-Démarrage de l'application FastAPI
-- router.py	-Gestion des routes de l'API
-- weather.py	-Récupération des données météo
-- speech_to_text.py	-Intégration Azure Speech-to-Text
-- processing.py	-Traitement des commandes vocales
-- static/	-Contient les fichiers frontend (CSS, JS)
-- templates/	-Contient les fichiers HTML (Jinja2)
-- prometheus.yml	-Configuration Prometheus pour le
monitoring	
-- requirements.txt	-Liste des dépendances Python

## 4 - Utilisation de l'Application

### Accès

- Ouvrir l'application sur <http://127.0.0.1:8000>
- L'utilisateur doit cliquer sur le bouton micro pour activer la reconnaissance vocale

### Commandes disponibles

- "Je veux la météo de Tours" → Affiche la météo du jour et des prochains jours
- "Quelle est la météo à Tours aujourd'hui ?" → Affiche la météo du jour
- "Donne-moi la météo de Tours pour 7 jours" → Affiche les prévisions sur 7 jours.

## Fonctionnalités

- Reconnaissance vocale (Azure Speech-to-Text)
- Prédiction automatique des villes et dates (Stanza + regex)
- Visualisation des données météo (Open-Meteo API)
- Monitoring de la qualité de l'application (Prometheus + Grafana)