Voice Automation Agent – Project Documentation

# 1. Project Overview

The Voice Automation Agent is an intelligent voice-driven system designed for appointment management through natural language interaction. It leverages advanced speech recognition, language understanding, and text-to-speech technologies to enable seamless human-like scheduling experiences.

# 2. Key Features

* 🎙️ Voice Recognition using OpenAI Whisper for accurate speech-to-text conversion.
* 🧠 Natural Language Understanding powered by Claude 3.5 Sonnet for intent extraction.
* 🔊 Text-to-Speech using Google TTS for realistic voice responses.
* 📅 Appointment Management with local storage and external API sync.
* 🗓️ Calendar Integration supporting ICS export and multi-calendar sync.
* 🧩 Modular Architecture for scalability and maintainability.

# 3. Tools & Technologies

* Programming Language: Python 3.10+
* Speech-to-Text: OpenAI Whisper
* LLM: mistral-7b-instruct (via OpenRouter API)
* Text-to-Speech: Google TTS
* Frameworks/Libraries: dotenv, json, requests, datetime
* Data Storage: JSON-based local storage for schedules and calendar events

# 4. System Architecture

The system follows a modular architecture to ensure separation of concerns and ease of maintenance:

* speech\_io.py – Handles voice input and TTS output.
* llm\_interface.py – Manages LLM communication and intent detection.
* scheduler.py – Manages local appointments and task scheduling.
* calendar\_integration.py – Handles calendar event creation and export.
* validation.py – Validates input formats for time and dates.
* logger.py – Maintains logs for debugging and traceability.

# 5. 4-Day Action Plan

|  |  |  |
| --- | --- | --- |
| Day | Focus Area / Goal | Summary of Tasks |
| Day 1 | Research & Planning | Researched voice automation systems, studied Whisper, Mistral, and Google TTS. Defined objectives and designed project structure. |
| Day 1 | System Architecture Design | Outlined data flow between modules and decided LLM integration strategy. |
| Day 2 | Environment Setup & Core Implementation | Configured virtual environment, dependencies, and implemented Whisper-based voice recognition. |
| Day 2 | Voice Interaction Loop | Developed end-to-end voice → LLM → TTS pipeline and tested transcription accuracy. |
| Day 3 | Scheduling & Integration | Integrated local appointment management and external calendar syncing. |
| Day 3 | Debugging & Testing | Performed unit and integration tests, fixed bugs, and ensured smooth workflow. |
| Day 4 | Documentation & Presentation | Finalized README, prepared demo and presentation slides. |

# 6. Testing & Validation

Comprehensive testing ensured robust performance and reliability across components:

* ✅ Unit Tests for speech recognition, scheduling, and validation.
* 🔁 Integration Tests for voice-to-LLM-to-TTS loop.
* 📊 System Tests covering end-to-end appointment management workflows.

# 7. Conclusion

The Voice Automation Agent successfully demonstrates how AI-driven voice technology can simplify appointment management. Its modular architecture allows future extensions, including multi-language support, advanced dialogue management, and integration with real-time calendar APIs.