**AI Character Assistant Project**

**Working Demo**

**Deployed Application or Local Setup**

The AI Character Assistant is designed as a Streamlit application. It can be run locally or deployed on cloud platforms such as Streamlit Cloud or AWS or hugging face. Upon launching, the application allows users to interact with a custom AI character capable of engaging in text and voice-based conversations.

**Sample Conversations**

* **User:** "How are you today?"
* **AI Assistant:** "Hey there! I'm doing great. How can I assist you today?"
* **User (via voice):** "Tell me something about AI."
* **AI Assistant:** "AI stands for Artificial Intelligence. It's a fascinating field where machines are programmed to mimic human intelligence!"

**Character Background**

* **Name:** Luna
* **Age:** 25
* **Gender:** Female
* **Interests:** Technology, Science,art, etc
* **Communication Style:** Casual or professional, depending on configuration.

Luna is a friendly AI assistant who adapts to user inputs and maintains a conversation style consistent with her configured personality.

**Feature Demonstration**

1. **Voice Input**: Users can speak directly to the assistant, and the app transcribes the speech into text for processing.
2. **Text-to-Speech Output**: The assistant’s responses can be played back using a voice that matches the character's gender.
3. **Dynamic Emotional State Management**: The assistant’s responses adapt based on the emotional tone of user inputs.
4. **Customizable Character**: Users can configure the assistant’s name, age, gender, personality traits, and interests.
5. **Memory**: The assistant remembers the last few interactions for contextual continuity.

**Technical Documentation**

**Architecture Overview**

The project follows a modular design, integrating multiple technologies for voice processing, text interaction, and AI character modeling.

1. **Frontend**:
   * Streamlit provides the user interface for chat interactions and customization.
2. **Backend**:
   * Google Generative AI for conversation modeling.
   * Speech recognition and synthesis via speech\_recognition and pyttsx3.
3. **Middleware**:
   * Personality and emotional state engine to adapt responses dynamically.

**System Components**

1. **AI Character**:
   * A personality-driven engine based on a CharacterProfile data class.
   * Emotional state adjustment based on user inputs.
2. **Conversation Memory**:
   * Stores short-term and long-term memories for contextual understanding.
3. **Voice Processing**:
   * **Input**: Speech-to-text using speech\_recognition.
   * **Output**: Text-to-speech with configurable speed and voice settings.
4. **Generative AI Integration**:
   * Leverages Google’s Gemini 1.5 model for generating context-aware responses.

**AI Model Integration**

* **Model**: Google Gemini 1.5 Pro
* **Interaction**: Prompts are dynamically crafted based on the character’s personality and recent context from memory.

**Data Flow Diagrams**

1. **Voice Input**:
   * User speaks → Audio captured by microphone → Converted to text → Processed by AI → Response displayed and played back.
2. **Text Interaction**:
   * User types a message → Input processed by AI → Response displayed → Optionally converted to speech.

**Future Expansion Possibilities**

1. **Enhanced Personalization**:
   * Allow users to save and load character configurations for tailored experiences.
2. **Integrations**:
   * Add support for external APIs (e.g., calendar, email, or task management).
3. **Character Development:**

* Using readplayme or blender integration along with character lip-sync and character movements.

This project demonstrates the fusion of conversational AI, emotional intelligence, and customizable experiences, passing the way for more engaging and human-like interactions.