Project Documentation: OurCode

Team Name:

OurCode

Team Members:

1. **Pedamallu Sai Mrudula** (Team Leader)
   Email: cb.sc.u4aie24340@cb.students.amrita.edu
2. **Meera S**
   Email: cb.sc.u4aie24133@cb.students.amrita.edu
3. **Meghana K**
   Email: cb.sc.u4aie24232@cb.students.amrita.edu

Project Overview:

In an era where sustainable living is increasingly vital, individuals often find it challenging to reduce their environmental impact. This project aims to develop a chatbot that provides personalized, location-specific advice on waste reduction, energy saving, and eco-friendly choices. By simplifying the journey towards sustainability, the chatbot will empower users to adopt habits that contribute positively to the environment.

Problem Statement:

As interest in sustainable living grows, many individuals face challenges in figuring out how to reduce their environmental impact. From managing waste to selecting eco-friendly products, the journey toward a more sustainable lifestyle can be confusing and overwhelming.

To address this, we propose creating a chatbot that guides users in adopting sustainable habits through personalized, location-specific advice on waste reduction, energy saving, and eco-friendly choices. The chatbot will provide easy-to-understand tips and step-by-step guidance, empowering users to make simple changes like reducing plastic use, cutting down on food waste, and saving energy.

Solution Overview:

The solution will integrate features such as:

- **Progress Tracking**: Users can track their sustainability journey with streaks and rewards.
- **Interactive Challenges**: Engaging activities to motivate users.
- **Pop-up Notifications**: Gentle reminders to encourage consistent engagement.

Technical Aspects:

- **Natural Language Processing**: Utilizing Rasa for effective user interaction.
- **Backend Operations**: Implementing FastAPI to manage backend processes.
- **User Interface**: Developing an interactive and visually appealing UI.
- **Voice Integration**: Enhancing user experience with voice commands and responses.
- **API Connections**: Providing location-specific insights and eco-friendly product recommendations through external APIs.

By making sustainability easier to understand and take action on, this chatbot will inspire individuals to adopt meaningful habits that protect the environment.

Tech Stack:

- **Rasa Framework**: For building conversational AI and chatbots.
- **React.js**: To create dynamic user interfaces for the application.
- **Python Speech-to-Text**: For converting spoken language into text.
- **Speech Recognition**: To recognize and process speech commands.
- **Firebase**: For real-time database and user authentication.
- **Pinecone**: For vector database and managing AI data.
- **Custom Tkinter**: To create GUI applications in Python.
- **PIL (Python Imaging Library)**: For image processing tasks.
- **Matplotlib**: For data visualization in graphs and charts.
- **Pyplot**: Part of Matplotlib for plotting graphs easily.
- **Word Cloud**: For generating word clouds from text data.
- **Calendar**: To manage and display calendar-related functionalities.
- **Threading**: For managing multiple threads in the application.
- **ChatUI**: For creating an interactive chat interface.
- **Pygame**: For any game-related functionalities or graphics.
- **VoiceInput**: For handling input through voice commands.
- **Emotion Detection**: To analyze and detect emotions in user interactions.
- **Utils**: For utility functions to support various features.
- **FER (Facial Expression Recognition)**: For detecting facial emotions.

- **Flask**: To set up a web server for the application.
- **Transformers**: For utilizing pre-trained models in natural language processing.
- **PyAudio**: For audio handling and processing.

Project Features:

- **Conversational AI**: A chatbot that can interact with users based on their inputs.
- **Voice Commands**: Users can control the application through voice.
- **Emotion Recognition**: Analyzing user emotions for better interactions.
- **Progress Tracking**: Users can track their sustainable habits and earn rewards.
- **Interactive Challenges**: Fun activities to encourage eco-friendly behavior.
- **User-Friendly Interface**: Designed using React.js and Tkinter for ease of use.
- **Real-time Data**: Storing and retrieving data using Firebase.
- **Personalized Tips**: Location-specific advice tailored to user needs.

Future Enhancements:

- **Machine Learning Integration**: Incorporating more advanced machine learning algorithms to improve personalization and predictions.
- **Community Features**: Allowing users to share tips and experiences with others to create a community around sustainability.
-

Challenges Faced:

- **Data Collection**: Gathering reliable and relevant data for location-specific recommendations.
- **User Engagement**: Maintaining user interest and ensuring consistent usage over time.
- **Technical Integration**: Ensuring all components (Rasa, FastAPI, Firebase) work seamlessly together.