The Empathy Engine (Core Solution)

A quick-turnaround prototype for the **Challenge 1: The Empathy Engine** hackathon, focusing on delivering all **Core Functional Requirements** using a simple, self-contained Python/Flask stack.

Core Solution Details

This solution successfully implements all five **Must-Have** requirements:

Requirement	Implementation
1. Text Input	Provided via a simple Flask web form.
2. Emotion Detection	Uses the **VADER** sentiment library to classify text into **Positive**, **Negative**, and **Neutral** categories.
3. Vocal Modulation (2 Params)	Modulates the **Rate** (speed) and **Volume** of the TTS output.
4. Emotion-to-Voice Mapping	A clear, demonstrable logic is implemented in `app.py`.
5. Audio Output	Generates a playable `.mp3` file using the offline `pyttsx3` engine.

Setup and Deployment Instructions

This application is designed to run locally using the provided files.

1. Folder Structure

Ensure your file structure matches this layout:

empathy-engine/

```
├── app.py
├── requirements.txt
├── README.md
├── static/
│   └── audio/  <- (Automatically created by app.py)
└── templates/
    └── index.html
```

2. Environment Setup

Create a Virtual Environment (Recommended):
python -m venv venv
source venv/bin/activate # On Windows: venv\Scripts\activate

1.

Install Dependencies:
pip install -r requirements.txt

3. Run the Application

python app.py

1. **Access the Interface**: Open your browser and navigate to the address shown
   (usually `http://127.0.0.1:5000/`).

Design Choices: Emotion Mapping Logic

Detected Emotion (VADER Score)	Contextual Goal	Rate (WPM)	Volume (0.0-1.0)
Positive (Score > 0.2)	**Enthusiasm/Excitement**	**200** (Fast)	**1.0** (Max)
Negative (Score < -0.2)	**Seriousness/Patience**	**120** (Slow)	**1.0** (Max)
Neutral (Score between -0.2 & 0.2)	**Information Transfer**	**150** (Standard)	**1.0** (Max)

- **Positive:** A faster rate conveys excitement and energy for good news.

- **Negative:** A slower, measured rate is used to convey patience and a calm, serious tone when addressing a customer's frustration (avoiding an angry or rushing voice).
- **Rate & Volume Parameters:** The solution uses **Rate** (words per minute) and **Volume** (amplitude) as the two distinct vocal parameters that are programmatically altered based on the detected emotion.