EchoChat TTS Backend - Local Setup Guide

This guide helps you run the Text-to-Speech (TTS) backend of the EchoChat project locally, using Python and Ngrok for tunneling.

Prerequisites

- Python 3.x installed
- Virtual environment set up (with all dependencies inside `envtts`)
- Ngrok installed and connected to your account
- Node.js (optional for frontend testing)

Step 1: Start the Python TTS Server

---

cd ~/Documents/chatBot/tts

source envtts/bin/activate

python tts_api.py

Expected Output:

Running on all addresses (0.0.0.0)

 * Running on http://127.0.0.1:5050

 * Running on http://192.168.1.36:5050

Press CTRL+C to quit

 * Restarting with stat

* Debugger is active!

* Debugger PIN: 104-133-329

Step 2: Tunnel Your Local Server via Ngrok

In a new terminal, run:

```
ngrok http 5050
```

Expected Output:

```
Session Status                online
Account                       Akash Verma (Plan: Free)
Version                       3.22.1
Region                        India (in)
Web Interface                 http://127.0.0.1:4040
Forwarding                    https://9434-192-140-153-103.ngrok-free.app -> http://localhost:5050
```

Note: The https://...ngrok-free.app link is your public TTS API URL

Step 3: Run the Frontend (Optional)

Start your frontend locally:

```
npm start
```

Or use the deployed frontend:

https://chatbot-ten-sigma-93.vercel.app

Your app is now connected to the live TTS server!

Notes

- Replace all internal localhost:5050 references with the new ngrok URL
- Ngrok URLs change per session unless you're on a paid plan
- If deploying to Vercel, update environment variables accordingly

---

Maintainer

Akash Verma

Email: [Your Email]

GitHub: [Your GitHub URL]