Live Transcription Device Manual

**Introduction**:

The Live Transcription Device is designed to take speech from a user and turn it into text that will be displayed on a screen using vosk. This manual will go over the specs of the device, how to connect them, what software is used to, and how the devices boot into the app.

**Hardware Specifications**:

The device uses the following components:

* **Raspberry Pi 4 Model B**: This component is the computer for the device handling all the data processing, The specs for this component can be found on the raspberry pi website (<https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>). The device requires 5V of power via USB-C.
* 11.9 inche waveshare LLCD screen: This component acts as the display for the device. The resolution for the screen is 320x1480. Power is received via micro-USB is displays output via HDMI. Four small Philp screws come with it to attach the Pi to the screen
* Maono AU-UL10 USB Lavalier Microphone: This component is an omni-directional microphone which means that it picks up sound from every direction. It is powered via USB.
* 16 GB < microSD card: The micro SD card stores the operating system and the application and is executed on the raspberry pi.
* USB-C to 5V power supply: needed to turn on the device
* Micro-HDMI to HDMI connector: This is for the raspberry pi to display output to the screen.
* PLA plastic enclosure

**Assembly:**

The assembly of this device can be done in any order if all the components are connected properly.

1. Insert the Micro SD card into the raspberry pi. The slot for it is on the bottom of the raspberry pi

A green circuit board with many small holes

AI-generated content may be incorrect.

1. The raspberry pi board is the be screwed into the back of the waveshare LCD screen via 4 Philips screws.

A green electronic device with many different components

AI-generated content may be incorrect.

1. The LCD screen is then powered by a micro-USB to USB-B cable. The micro-USB end connects to the LCD screen and the USB-B end connects to the top USB 2.0 port on the raspberry pi which is indicated by a black port.

A close-up of a circuit board

AI-generated content may be incorrect.

A close up of a circuit board

AI-generated content may be incorrect.

1. Using the HDMI to micro HDMI connector. Insert the HDMI end into the waveshare screen and the micro HDMI end into the raspberry pi.

A close up of a circuit board

AI-generated content may be incorrect.

1. The microphone is then connected to the raspberry pi by USB-B. to connect it insert the USB-B part into the bottom USB-3.0 port which is indicated by a blue port. Make sure the cable goes through the enclosure.

A computer board with wires

AI-generated content may be incorrect.

A close up of a device

AI-generated content may be incorrect.

1. Insert the device into the enclosure through the front of the enclosure.

A black rectangular object with a black wire on a wooden surface

AI-generated content may be incorrect.

1. To the power the entire device. A USB-C power supply must be used. This power supply provides ~5V to the device. Plug the USB-C end into the raspberry pi and the plug into the wall.

A close up of a computer

AI-generated content may be incorrect.

**Software Requirements:**

The following are the required software needed to run the device

* Raspberry Pi OS Bullseye Lite: The device needs to use raspberry pi os bullseye for the raspberry pi to output anything to the screen. The lite version is the headless version of the os.
* Python 3.9.2: This version of python comes preinstalled with bullseye
* Vosk: A transcription model design for low power systems. Models can be found here (<https://alphacephei.com/vosk/models>)
* Joes window manager (jwm): a light window manager that allows for graphical user interfaces to be displaced. To install use the command “sudo apt install jwm”
* PortAudio: this is a driver that allows the raspberry pi to record audio using a micro phone. To install it use the command “sudo apt-get install portaudio19-dev”
* Lightdm: this will allow you to boot into jwm. To install use “sudo apt install lightdm”

Python dependencies:

* Sounddevice: This library takes advantage of the Port audio driver to record sound on the raspberry pi. The sampling rate is set to 16 kHz with a block size of 1024 frames per block. It’s important to configure these 2 as the more blocks/higher rate you have. The more CPU resources you will need to use. But too low can cause poor audio quality.
* word2number: depend
* Vosk: this library loads in the model and transcribes whatever the user has said.
* Tkinter: this is a GUI framework that comes bundled with the raspberry pi. It is very lightweight and fast.

**How to start the device:**

The device automatically starts once plugged in and the app runs on start-up via the . jwmrc file in the home directory. This file is an XML file that tells jwm what to load when starting up. To disable this, you will need to edit the ~/.jwmrc file and disable the startup command at the bottom of the file. It should look like this “<StartupCommand>/home/pi/AI-chatbot/start.sh &</StartupCommand>”. This can be done using vim or nano which are both installed on the device.

**How to connect to the device from laptop or computer:**

To connect to the device and modify the software you will need a CAT-5 or CAT-6 cable which will connect to both . Once the device as been connected to the laptop you should be able to connect to the device by using ssh. The command for that would be “ssh [pi@raspberrypi.local](mailto:pi@raspberrypi.local)”. Here pi is the username of the device and raspberrypi.local is its address. If this is your first time it will prompt you to say yes or no. type yes. After it will ask for a password. The password is “AIChatbot”. After you will be connected via ssh. You can also connect via vnc server if you have real vnc server installed on your laptop. To copy files from the device or to the devices you can use:

* Scp filepath [pi@raspberrypi.local:directory](mailto:pi@raspberrypi.local:directory)path (to the device)
* Scp [pi@raspberrypi.local:file](mailto:pi@raspberrypi.local:file)path directorypath (from the device)