

GitaGPT Hardware Setup Guide

Server Hardware (Windows/Linux)

Minimum Requirements

- **CPU:** Intel i5 or AMD Ryzen 5 (4+ cores)
- **RAM:** 8GB (16GB recommended for better performance)
- **Storage:** 10GB free space for models and data
- **OS:** Windows 10/11 or Ubuntu 18.04+
- **Network:** WiFi or Ethernet for client communication

Recommended Specifications

- **CPU:** Intel i7 or AMD Ryzen 7 (8+ cores)
- **RAM:** 16GB+ for optimal Whisper and FAISS performance
- **GPU:** Optional - CUDA-compatible GPU for Whisper acceleration
- **SSD:** For faster model loading

Client Hardware (Raspberry Pi)

Required Components

- **Board:** Raspberry Pi 4B (4GB RAM minimum, 8GB recommended)
- **SD Card:** Class 10, 32GB+ for OS and dependencies
- **Power Supply:** Official 5.1V 3A USB-C adapter
- **Case:** With fan or heatsink for thermal management

Audio Components

- **Microphone:** USB microphone or USB sound card with 3.5mm input
- **Speakers:** USB-powered speakers or 3.5mm output to amplifier
- **Alternative:** USB headset for combined mic/speakers

Network

- **WiFi:** Built-in WiFi (ensure good signal to server)
- **Ethernet:** Optional but recommended for stable connection

Arduino (Humanoid Version)

Arduino Board

- **Model:** Arduino Uno, Nano, or compatible
- **USB Cable:** For connection to Raspberry Pi
- **Power:** Can be powered via USB from Pi

Servo Motor

- **Type:** Standard servo (SG90 or similar)
- **Voltage:** 5V compatible
- **Torque:** Sufficient for jaw mechanism (2-3 kg·cm)
- **Connection:** 3-wire servo cable to Arduino

Mechanical Components

- **Servo Horn:** Attached to jaw mechanism
- **Mounting:** Secure servo to robot head/jaw assembly
- **Linkage:** Mechanical connection from servo to jaw

Wiring Diagrams

Arduino Servo Connection

Arduino Uno	Servo Motor
GND	Brown/Black wire
5V	Red wire
Pin 9	Orange/Yellow wire (Signal)

Raspberry Pi Connections

Pi 4B	Component
USB ports	→ Arduino (USB cable)
USB ports	→ Microphone
3.5mm jack	→ Speakers
GPIO (optional)	→ Status LEDs

Assembly Notes

Arduino Programming

Upload the servo control sketch before connecting:

```
// Simple servo control for GitaGPT jaw movement
#include <Servo.h>

Servo jawServo;
int servoPin = 9;

void setup() {
    Serial.begin(9600);
```

```

jawServo.attach(servoPin);
jawServo.write(90); // Neutral position
}

void loop() {
    if (Serial.available()) {
        char command = Serial.read();
        switch(command) {
            case 'O': // Open jaw
                jawServo.write(180);
                break;
            case 'c': // Close jaw
                jawServo.write(0);
                break;
            case 's': // Stop/neutral
                jawServo.write(90);
                break;
        }
    }
}

```

Power Considerations

- Pi 4B: 15W (3A @ 5V)
- Arduino: 2W (powered via USB from Pi)
- Servo: 1-2W during movement
- Total: ~18W maximum

Cooling

- Ensure adequate ventilation for Pi under continuous operation
- Consider fan or heatsink if running intensive models
- Monitor CPU temperature: `vcgencmd measure_temp`

Troubleshooting Hardware Issues

Audio Problems

- **No microphone input:** Check USB connection, verify with `lsusb`
- **No audio output:** Test with `speaker-test -t wav`
- **Poor audio quality:** Check sample rate settings (16kHz)

Arduino Issues

- **Not detected:** Check USB cable, try different ports
- **Servo not moving:** Verify power supply, check wiring
- **Random movements:** Check serial baud rate (9600)

Network Connectivity

- **Client can't reach server:** Verify IP address, check firewall
- **Slow response:** Consider wired Ethernet connection
- **Intermittent connection:** Check WiFi signal strength

Performance Issues

- **Slow Whisper:** Reduce model size (tiny → base → small)
- **High CPU:** Monitor with `htop`, consider server hardware upgrade
- **Memory errors:** Close unnecessary applications, add swap

Hardware Testing Commands

Audio Testing (Pi)

```
# Test microphone
arecord -f cd -d 5 test.wav
aplay test.wav

# Test speakers
speaker-test -t wav -c 2

# Check audio devices
aplay -l
arecord -l
```

Arduino Testing

```
# Find Arduino port
ls /dev/tty*

# Test serial communication
screen /dev/ttyUSB0 9600
# Type: 0, c, s commands
```

Network Testing

```
# Test server connectivity
ping 192.168.1.100
curl http://192.168.1.100:5000/health
```