RENTAHAL Windows Setup Guide: Complete Installation Instructions

Prerequisites

Before starting, ensure you have a Windows machine with a dedicated NVIDIA GPU.

Step 1: CUDA Installation

1.1 Install NVIDIA Drivers

1. Download the latest NVIDIA driver from https://www.nvidia.com/Download/index.aspx
2. Run the installer and follow the prompts
3. Restart your computer

1.2 Install CUDA Toolkit

1. Go to https://developer.nvidia.com/cuda-downloads
2. Select Windows → x86_64 → Select your Windows version
3. Download and run the installer (approximately 3GB)
4. During installation:
   - Choose "Express Installation"
   - Wait for completion (10-15 minutes)
5. Verify installation by opening Command Prompt and typing:

   ```bash
   nvcc --version
   ```

Step 2: Ollama Installation

2.1 Install Ollama for Windows

1. Visit https://ollama.ai/download
2. Click on "Download for Windows"
3. Run the `OllamaSetup.exe` installer
4. Follow the installation wizard
5. Ollama will automatically start as a service

2.2 Pull Required Models

Open Command Prompt and run:

```bash
# Pull LLaMA 3 model (approximately 4GB)
ollama pull llama3

# Pull LLaVA model (approximately 5GB)
ollama pull llava
```

Step 3: Python Environment Setup

3.1 Install Python

1. Download Python 3.11 from https://www.python.org/downloads/
2. During installation:
   - Check "Add Python to PATH"
   - Click "Install Now"

3.2 Create Virtual Environment

Open Command Prompt in your project directory:

```bash
# Navigate to your project directory
cd C:\path\to\your\rentahal\project

# Create virtual environment
python -m venv venv

# Activate virtual environment
venv\Scripts\activate

# Install required packages
pip install fastapi uvicorn pydantic pillow torch torchvision torchaudio --index-url https://dc
```

Step 4: Setting Up FastAPI Workers

4.1 Create Required Files

Ensure you have these files in your project directory:

- `main.py` (the FastAPI application)
- `llama-fast.cmd` (for starting LLaMA worker)
- `llava-fast.cmd` (for starting LLaVA worker)

4.2 Modify Worker Scripts

Edit your `.cmd` files to match your environment:

llama-fast.cmd:

```batch
@echo off
cd C:\path\to\your\rentahal\project
call venv\Scripts\activate
start uvicorn main:app_llama --host 0.0.0.0 --port 8000 --log-level debug
```

llava-fast.cmd:

```batch
@echo off
cd C:\path\to\your\rentahal\project
call venv\Scripts\activate
start uvicorn main:app_llava --host 0.0.0.0 --port 8001 --log-level debug
```

Step 5: Starting the Workers

5.1 Launch Workers

1. Open two separate Command Prompt windows
2. In the first, run: `llama-fast.cmd`
3. In the second, run: `llava-fast.cmd`

5.2 Verify Workers

Test workers using curl or a browser:

```bash
# Test LLaMA health
curl http://localhost:8000/health

# Test LLaVA health
curl http://localhost:8001/health
```

Step 6: Stable Diffusion Setup

6.1 Install Stable Diffusion Web UI

1. Install Git from https://git-scm.com/download/win
2. Open Command Prompt:

```bash
# Navigate to your preferred installation directory
cd C:\ai-models

# Clone the repository
git clone https://github.com/AUTOMATIC1111/stable-diffusion-webui.git

# Navigate to the directory
cd stable-diffusion-webui

# Run the installation
webui-user.bat
```

6.2 Configure for API Access

Edit `webui-user.bat` to enable API:

```batch
@echo off

set PYTHON=
set GIT=
set VENV_DIR=
set COMMANDLINE_ARGS=--api --listen

call webui.bat
```

6.3 Download Stable Diffusion 1.5 Model

1. Visit https://huggingface.co/runwayml/stable-diffusion-v1-5
2. Download `v1-5-pruned.ckpt` (4GB)
3. Place in `stable-diffusion-webui\models\Stable-diffusion\`
4. Restart the web UI

6.4 Verify API Access

Test the API endpoint:

```bash
curl http://localhost:7860/sdapi/v1/txt2img
```

Step 7: Connecting RENTAHAL to Workers

Update your RENTAHAL configuration to point to the workers:

1. In `webgui.py`, update worker addresses:

```python
# Default worker configuration
DEFAULT_WORKER_ADDRESS = 'localhost:8000'  # LLaMA worker
```

2. Add workers to the database:

```python
# Add via admin panel or API
workers = [
    {'name': 'llama_worker', 'address': 'localhost:8000', 'type': 'chat'},
    {'name': 'llava_worker', 'address': 'localhost:8001', 'type': 'vision'},
    {'name': 'sd_worker', 'address': 'localhost:7860/sdapi/v1', 'type': 'imagine'}
]
```

Step 8: Testing the Complete System

1. Start all services:
   - Run `llama-fast.cmd`
   - Run `llava-fast.cmd`

- Run `webui-user.bat` (for Stable Diffusion)
- Start your RENTAHAL webgui.py

2. Test endpoints:
   - Chat: Should route to LLaMA worker
   - Vision: Should route to LLaVA worker
   - Imagine: Should route to Stable Diffusion

Troubleshooting

Common Issues:

1. **CUDA not found**: Ensure NVIDIA drivers are up to date
2. **Ollama models stuck**: Check internet connection and disk space
3. **Workers not starting**: Verify paths in `.cmd` files
4. **Stable Diffusion out of memory**: Reduce batch size or resolution

Log Files to Check:

- Worker logs: Look in Command Prompt windows
- RENTAHAL logs: Check `webgui_detailed.log`
- Stable Diffusion logs: Check SD web UI output

Maintenance Tips

1. **Update Ollama regularly**:

   ```bash
   ollama pull llama3
   ollama pull llava
   ```

2. **Update CUDA and drivers** periodically for performance
3. **Monitor GPU usage** with NVIDIA System Monitor
4. **Backup your models** and configurations

System Requirements

- Windows 10/11
- NVIDIA GPU with 8GB+ VRAM
- 16GB+ System RAM

- 50GB+ Free disk space
- Fast internet connection for model downloads

Conclusion

Your RENTAHAL system is now ready with:

- LLaMA for text generation
- LLaVA for image analysis
- Stable Diffusion for image generation

All workers expose HTTP/REST APIs that RENTAHAL can access through its modular architecture.