Personal Project

PR-07

Homelab

Your Internet

Project Neuromancer

Gonzalo Torras Serrano

Table of Contents

List of Tables

1. Description of the project

Project Neuromancer is a private AI HomeLab designed to create a capable and responsive local assistant. The goal is to build a system that runs entirely on-premise to provide helpful automation, coding assistance, and media management without relying on external cloud subscriptions.

The system uses a hybrid architecture that combines a powerful main server with efficient low-power devices. A central PC equipped with an RTX 3090 handles the heavy processing tasks like running the Qwen 3 language model for reasoning and voice interaction. This main node is supported by a cluster of Raspberry Pi devices that handle essential background tasks like network security and home automation around the clock.

The design follows a strict separation of duties. The high-performance server focuses on AI and media tasks while the smaller devices ensure the network and security systems stay online reliably. This setup allows the assistant to offer personalized help with studies and projects using local data while keeping the home network stable and secure.

Project Neuromancer aims to be a practical and evolving tool. It learns from daily usage to become more effective at managing schedules, organizing information, and controlling smart home devices, all while keeping personal data within the local network.

2. Project Scope & Objectives

2.1 Core Objectives

Privacy and Sovereignty

The primary goal is to keep all personal data within the local network. Financial records, personal notes, and voice conversations must be processed and stored on-premise to ensure no information is shared with third-party providers.

Offline Resilience

The system is designed to maintain critical functionality without an internet connection. Essential services like DNS resolution, home automation, and security systems must operate independently of the main AI server status.

Progressive Customization

The AI assistant will evolve over time using Retrieval-Augmented Generation (RAG) and specific LoRA adapters. This allows the system to adapt to new tasks and learn from the user's coding projects and academic notes without altering the base model.

High-Performance Integration

The architecture aims to support demanding workloads such as 4K media streaming and PC gaming alongside AI tasks. The system must manage these resources dynamically to ensure smooth performance across all activities.

2.2 Functional Requirements

Local Voice Control

A responsive voice interface must be available in key rooms. It should support wake-word detection and natural language with low latency to control devices and answer queries.

Intelligent Automation

The AI agent requires read and write access to calendars, to-do lists, and IoT states. It should be capable of proactively managing schedules and executing complex home automation routines based on context.

Automated Media Management

The system must handle the organization of the personal media library and the ingestion of legally obtained content automatically.. It should provide a seamless viewing experience with support for offline synchronization to mobile devices for travel.

Comprehensive Security

The infrastructure must include robust network security measures. This includes network-wide ad-blocking, a secure VPN for remote access, and a centralized identity management system for all internal services.

3. System Architecture & Topology

3.1 Compute Purity Philosophy

The system architecture enforces a strict separation of services based on resource usage and uptime requirements. This approach ensures stability and efficient resource allocation across the network.

Tier 1 - The Monolith (High-Compute)

This tier consists of the RTX 3090 server. It is dedicated exclusively to heavy workloads that require significant processing power, such as AI inference, media transcoding, and gaming. Since these tasks are resource-intensive, this node is treated as an on-demand resource that can be rebooted or suspended for gaming without affecting the home's basic connectivity.

Tier 2 - The Infrastructure Hub (Critical Edge)

This tier runs on the Raspberry Pi 5. It hosts essential 24/7 services that must remain online at all times. This includes the network core, DNS, and security vaults. By offloading these tasks from the main server, the home network remains functional and secure even if the high-compute node is offline.

Tier 3 - The Sensory Edge (Passive)

This tier is composed of distributed Raspberry Pi Zero nodes. Their sole function is audio capture and simple presence detection. They are designed to be wireless and low maintenance to allow flexible placement in different rooms without complex cabling.

3.2 Network & Security Strategy

Physical Layer

The backbone of the network relies on a Gigabit Ethernet connection for the main servers. An unmanaged switch is used to expand port capacity for wired devices to ensure stable high-speed data transfer between the Monolith and the Hub.

Access Control

Identity management is centralized using Authentik to provide Single Sign-On (SSO) capabilities. This simplifies access to dashboards while enforcing stricter security policies for sensitive areas. Critical services like the password manager require Two-Factor Authentication (2FA) for entry.

Resilience

Power stability is managed by a 1500VA UPS connected to the main server. It integrates with Network UPS Tools (NUT) to trigger automated graceful shutdowns during power outages to prevent data corruption. Data protection follows a 3-2-1 backup strategy using BorgBackup to create daily snapshots on a local external hard drive attached to the Infrastructure Hub.

4. Hardware Infrastructure Specifications

4.1 Node 1: The Monolith (AI & Media Server)

This node serves as the computational backbone of the system. The hardware has been

In previous iterations of this HomeLab, workloads were often scattered across older hardware or repurposed consumer laptops, which led to significant bottlenecks when attempting to run modern AI models or stream high-quality media. The lack of dedicated VRAM meant that Large Language Models (LLMs) had to run on the CPU, resulting in painfully slow response times, while gaming or transcoding would frequently crash the system due to thermal throttling.

To overcome these limitations and build a true "Enterprise-Grade" home assistant, the new architecture centers around a purpose-built high-performance server. This machine is designed specifically to eliminate those bottlenecks, providing a massive buffer of video memory for AI and enough raw processing power to handle multiple concurrent Docker containers without breaking a sweat. These are the technical specifications:

Processor (CPU) - Intel Core i7-13700K 3.4 GHz

A 16-core hybrid processor (Raptor Lake) capable of handling heavy multitasking. Its architecture allows background services to run on efficiency cores while performance cores remain dedicated to gaming or active inference.

Graphics Card (GPU) - ASUS TUF Gaming NVIDIA GeForce RTX 3090 OC (24GB)

The cornerstone of the AI stack. The 24GB of GDDR6X memory is essential for loading large, quantized LLMs (like Qwen 3) entirely into VRAM, ensuring instant reasoning capabilities.

Memory (RAM) - CORSAIR Vengeance RGB DDR5 64GB (2x32GB) 6200MHz

High-speed, high-capacity memory that facilitates rapid vector database retrieval and ensures smooth operation of memory-intensive applications like Jellyfin and databases.

Motherboard - ASUS TUF GAMING Z790 PLUS WIFI

Provides the stable power delivery and PCIe 4.0/5.0 connectivity required to fully utilize the speed of the NVMe drives and the GPU.

Cooling - Thermalright Peerless Assassin 120mm SE

A high-performance air cooler chosen to keep the i7-13700K operating at peak frequencies during sustained workloads.

Power Supply - Corsair RMe Series RM1000e (1000W) 80 Plus Gold

An ATX 3.1 compliant power supply ensuring stable energy delivery, specifically capable of handling the transient power spikes of the RTX 3090.

Primary Storage - Kingston NV3 2TB SSD M.2 NVMe PCIe 4.0

Dedicated "Hot Storage" for the Operating System, AI models, and active game files to maximize speed and responsiveness.

Secondary Storage - 20TB HDD

Dedicated "Cold Storage" for the automated media library (Movies/TV) and robust system backups.

4.2 Node 2: Infrastructure Hub (Critical Edge)

While the Monolith handles raw power, the Infrastructure Hub ensures reliability. This node runs on a Raspberry Pi 5 equipped with 8GB of RAM, a substantial upgrade that allows it to manage network traffic without latency. It utilizes an NVMe boot drive to ensure the operating system and critical databases respond instantly, alongside an external USB HDD dedicated to local backups. Connected directly to the router via Gigabit Ethernet, this device hosts the "always-on" services, such as Pi-hole for DNS, Wireguard for VPN access, and the Bitwarden vault, ensuring the smart home remains functional and secure even if the main server is offline.

4.3 Nodes 3-5: Sensory Satellites (Voice Interface)

The voice interface is distributed across three Raspberry Pi Zero 2 W units, chosen for their minimal footprint and wireless capabilities. Each unit is fitted with a ReSpeaker 2-Mic HAT to capture clear audio from anywhere in the room. These nodes operate entirely over Wi-Fi, eliminating the need to run Ethernet cables to bedrooms or living areas. Their role is strictly passive; they run lightweight wake-word detection software locally and stream audio to the Monolith for processing, acting effectively as the "ears" of the house.

4.4 Gaming Console

To separate leisure from critical infrastructure, a PlayStation 4 serves as the dedicated host for console-exclusive titles. By offloading these gaming workloads to a specific device, the main AI server preserves its resources for intelligence tasks, ensuring that the assistant remains responsive even during entertainment sessions.

4.5 Power Protection Unit

To guarantee data integrity across the ecosystem, a Tecnoware UPS ERA PLUS 1500 UPS (Uninterruptible Power Supply) is deployed as the first line of defense. This unit powers both the Monolith and the Infrastructure Hub. It is connected via USB to the Raspberry Pi 5, which runs a Network UPS Tools (NUT) server. In the event of a power outage, the UPS keeps the system running for approximately 15-20 minutes, enough time to smooth out brief fluctuations or, if the battery reaches a critical level, to trigger an automated, graceful shutdown command to all servers, preventing database corruption.

5. Software & Service Ecosystem

5.1 Artificial Intelligence Stack

The core intelligence of the system is built on a stack of open-source tools running on the main server. Ollama or vLLM serves as the backend to run the Qwen 3 model. This specific model is chosen for its balance of high reasoning capability and efficient memory usage.

The AI operates as a fully integrated System Agent with user-level privileges across the entire HomeLab ecosystem. Beyond passive retrieval, it possesses read/write access to internal APIs, allowing it to perform concrete actions: it can create and modify calendar events, edit Obsidian notes, curate Jellyfin playlists based on viewing history, and analyze financial data from Ivy Wallet to provide spending insights. It also interfaces with the notification system to send proactive alerts via Pushbullet and monitors container logs using the VPS and n8n access for self-diagnostic purposes.

To support this, a RAG Engine backed by a vector database like Qdrant manages long-term memory, while Function Calling capabilities enable access to live internet data for weather or sports. Crucially, access to these powerful tools is gated by multi-user voice identification. Verified users are granted full administrative control over personal data and automation, while unrecognized voices are restricted to a "Guest Profile," limited to safe, general knowledge queries.

For voice interaction, Whisper Server handles transcription. Once the AI formulates a response, Piper generates the spoken reply locally to ensure privacy and low latency.

5.2 Media Automation Stack

The entertainment system is designed to be fully automated. Jellyfin acts as the central media player. It organizes the library and handles transcoding to ensure movies play smoothly on any device, including creating offline copies for travel.

To manage content, Jellyseer provides a user-friendly interface where users can request new movies or TV shows. These requests are handled by Radarr and Sonarr, which monitor configured legal sources, retrieve the requested titles and organize them into the library without manual intervention. For gaming, Sunshine streams PC games from the server to other devices using the Moonlight client to allow low-latency play anywhere in the house.

5.3 Critical Infrastructure

Essential network and security services run on the Raspberry Pi 5 to ensure 24/7 availability independent of the main server. Pi-hole functions as a network-wide ad blocker and local DNS server, while Wireguard provides a secure VPN tunnel for remote access. To further harden security, Vaultwarden hosts the password vault locally, and Authentik manages identity via a single login portal for all applications. Finally, Nginx Proxy Manager handles SSL certificates, encrypting connections to all internal services.

5.4 Operational Tools

Managing this complex ecosystem requires robust tools. Portainer provides a graphical interface to manage all Docker containers across the different nodes. Homepage serves as the main dashboard for the user to access services and view system status at a glance.

Internal system health is tracked by Glances, which monitors resource usage in real time. Crucially, an External Watchdog runs on a private remote VPS hosting a self-hosted n8n instance. This automation platform executes sophisticated diagnostic workflows that remotely check connectivity, parse system logs for critical errors, and validate service responses. If specific failure patterns are detected, n8n triggers immediate alerts independent of the home network's status. Finally, a custom "Gaming Mode" script acts as a resource kill-switch. Triggered via a webhook from the dashboard, it instantly suspends AI and NVR containers to free up 100% of the GPU's VRAM for high-performance PC gaming sessions.

6. Implementation Roadmap

Phase 1: The Backbone (Infrastructure Node)

The initial phase focuses on establishing the critical network services that must be reliable and always online. This involves setting up the Raspberry Pi 5 with the core operating system and Docker. The primary goal is to deploy Portainer for management, Pi-hole for network-wide ad blocking, and Wireguard for secure remote access. Once connectivity is secured, Authentik will be configured to manage user identities, and Vaultwarden will be used. This phase ensures the "nervous system" of the house is functional before adding heavy compute loads.

Phase 2: The Library (Storage & Media)

With the network secure, the focus shifts to the high-performance server. This phase involves installing the physical hardware, including the 20TB hard drive. The media stack is then deployed, configuring Jellyfin for streaming and the automation suite comprising Radarr, Sonarr, and Jellyseer. At this stage, the backup strategy is also implemented. BorgBackup is configured to take daily snapshots of critical data from the Pi 5 and the main server to the local archive drive, ensuring data safety from day one.

Phase 3: The Brain (AI Core)

This phase activates the primary purpose of the Monolith node. The NVIDIA drivers and container toolkit are installed to unlock the power of the RTX 3090. The AI stack is then deployed, starting with Ollama and the Qwen 3 model. The Vector Database is initialized, and scripts are set up to ingest notes and documents for the RAG system. Finally, the "Gaming Mode" scripts are written and tested to ensure the GPU can dynamically switch between AI duties and gaming workloads without conflict.

Phase 4: The Senses (Voice & Vision)

The final phase brings the system to life by giving it eyes and ears. Frigate is deployed on the main server to handle security camera feeds with object detection. Simultaneously, the three Raspberry Pi Zero 2 W satellites are assembled and placed in key rooms. The wake-word software is tuned to the specific acoustics of the house, and the audio pipeline is integrated with Home Assistant. This completes the loop, allowing the user to speak to the house and receive intelligent, context-aware responses.

Phase 5: Contextual Presence & Multi-Device Casting

The final evolutionary step is to detach the assistant from voice-only feedback. This phase introduces Contextual Casting, allowing the AI to intelligently route visual information to the most relevant screen based on the user's location and request. For example, if a user asks for a recipe in the kitchen, the AI will automatically push the text to the refrigerator's display or a nearby tablet. Similarly, a request for "my monthly spending" would trigger a graph visualization sent directly to the user's smartphone via Pushbullet, rather than just reading numbers aloud. This creates a seamless, ambient computing experience where the digital assistant manifests visually wherever needed.

7. Conclusion

Project Neuromancer establishes a robust blueprint for a private, intelligent home ecosystem. By moving away from cloud dependencies, the system ensures that personal data remains strictly under user control while delivering capabilities that rival commercial assistants.

The hybrid architecture successfully balances power and efficiency. The separation of high-performance computing from critical infrastructure guarantees reliability, ensuring that essential home services remain online regardless of the AI workload. With a clear roadmap for implementation, the project is positioned to evolve from a hardware concept into a fully functional, learning assistant that adapts to the user's needs over time.

Appendix A: Hardware Assignment List

Server: The Monolith (RTX 3090)

Primary Function: High-Performance Compute & Media.

Category	Services / Containers	Function
Artificial Intelligence	qwen3-server (Ollama)	Main LLM Reasoning Engine
	whisper-server	Speech-to-Text Transcription
	piper-tts	Text-to-Speech Generation
	rag-engine (Qdrant)	Vector Database for Long-Term Memory
	stable-diffusion	Image Generation
Media & Gaming	jellyfin	Media Server (Movies/TV) with Transcoding
	navidrome	Dedicated Music Streaming
	sunshine	Game Streaming Host (Moonlight)
Surveillance	frigate	NVR with Object Detection
Automation	radarr, sonarr	Media Acquisition Managers
	jellyseer	Media Request Interface
System	gaming-mode-script	Resource Kill-Switch
	portainer-agent	Management Agent

Table 1: Appendix A - Main Server Monolith Services

Server: Infrastructure Hub (Raspberry Pi 5 - 8GB)

Primary Function: Critical 24/7 Services.

Category	Services / Containers	Function
Network & Security	pihole	Network-wide Ad-blocking & DNS
	wireguard	VPN Server
	nginx-proxy-manager	SSL Reverse Proxy
	authentik	Single Sign-On (SSO) Provider
Personal Cloud	vaultwarden	Password Manager
	filebrowser	Web-based File Manager
	ivy-wallet	Finance & Budget Tracker
	radicale	CalDAV/CardDAV Server
IoT & Automation	home-assistant	Main IoT Controller
	mosquitto	MQTT Broker
	meshtastic	LoRA Mesh Communication
Ops & Backup	borg-backup	Automated Local Snapshots
	uptime-kuma	Service Monitoring
	homepage	Unified User Dashboard

Table 2: Appendix A - Raspberry PI 5 Services

Satellites (RPi Zero 2 W)

Primary Function: Distributed Audio Interface.

Component	Specification	Function
Software	wyoming-satellite	Audio Capture & Playback Client
	porcupine	Local Wake Word Detection
Hardware	ReSpeaker 2-Mic HAT	Microphone Array

Table 3: Appendix A - Raspberry PI Zero Audio Interfaces

Network & Power Gear

Primary Function: Physical Backbone.

Component	Specification	Function
Switch	8-Port Gigabit (Unmanaged)	Network Expansion
Power	1500VA UPS / 900W	Battery Backup (USB-Monitored via NUT)

Table 4: Appendix A - Network and Power Services

Appendix B: Resources & Inspiration

Key Repositories

- **Local AI:** Ollama, Whisper.cpp
- **Voice Assistant:** Wyoming Satellite, Home Assistant Voice
- **Automation:** Radarr, Sonarr

Community & Guides

- **Hardware:** *Jeff Geerling's Raspberry Pi Guides* (YouTube/Blog)
- **Self-Hosting:** *Wolfgang's Channel* (YouTube), *TechnoTim* (YouTube)
- **AI Integration:** *NetworkChuck* (YouTube).
- **HomeLab:** Ardens (YouTube).

Project Influences

- **Concept:** J.A.R.V.I.S. (Iron Man) – Context-aware, proactive assistance.
- **Architecture:** "The Homelab Show" – Separation of compute vs. storage principles.