Project Proposal

High-Availability Raspberry Pi Cluster with Auto-Scaling and Live Monitoring

Project Type	Infrastructure / High-Availability Cluster / Open-Day Demo
Authors & Developers	Würth Eric, Simon Max
Status	Finished

1. PROJECT OVERVIEW

Description

This project designs and deploys a Raspberry Pi cluster that hosts a website with high availability, live monitoring and automatic scaling. The cluster is built as a live demonstration for the school's Porte Ouverte to showcase what students learn during the BTS Cloud Computing programme. Five Raspberry Pi all booting from microSD cards, working together to serve a single website through one virtual IP address provided by MetalLB. If any Pi is unplugged, the website stays online and a monitoring page built directly into the website immediately shows the node going offline while the remaining Pis absorb its traffic. Load balancing is fully automatic: Traefik, the reverse proxy built into K3s, distributes requests across the cluster without any manual intervention. When traffic increases, K3s automatically scales the number of running containers to handle the load.

2. PROJECT GOAL & VALUE

Goal

To build a working Raspberry Pi cluster that hosts a website with automatic failover, integrated live monitoring and auto-scaling, and to present it as an interactive demo during the Porte Ouverte to attract prospective students to the BTS Cloud Computing.

Problem Being Solved

Prospective students and their parents often struggle to understand what "cloud computing" really means. Abstract explanations of redundancy, load balancing and scaling are rarely convincing on open days. This serves as a visual artefact that makes these concepts immediately obvious and memorable.

Value

The project demonstrates practical skills in Linux administration, high-availability design, load balancing and networking — all core competencies of the BTS Cloud Computing. Once built, the cluster becomes a reusable teaching asset for future open days, classroom demonstrations and student lab work.

3. PROJECT SCOPE

In Scope

- Assembly and network setup of 5 Raspberry Pi 4 (all using microSD card storage).
- Configuration of K3s as the container orchestrator, with Traefik as the reverse proxy and load balancer, and MetalLB to provide a single virtual IP address for the cluster.
- Deployment of a website served simultaneously from all 5 Pis.
- Automatic failover: if a Pi is shut down or unplugged, MetalLB reassigns the virtual IP to a healthy node and traffic is redirected within seconds.
- Automatic load balancing and auto-scaling: Traefik distributes traffic across all available Pis, and K3s automatically adjusts the number of running containers as load changes or nodes leave and rejoin.
- Monitoring page built directly into the website, showing per-Pi online/offline status, CPU, RAM, temperature and which Pi is currently serving the visitor.
- Clearly labelled node status display for Porte Ouverte visitors.
- Documentation for setup, reset procedure and demo scenarios.

4. FUNCTIONAL REQUIREMENTS

- The cluster must host a demo website accessible through a single virtual IP address provided by MetalLB.
- If any single Pi becomes unavailable, the website must remain reachable within a few seconds.
- The monitoring page inside the website must reflect node state changes in near real-time.
- The monitoring page must display per Pi: online/offline status, CPU, RAM, temperature, and whether it is currently serving requests.
- The cluster must automatically adjust traffic distribution as load changes and as Pis join or leave the pool, without any manual intervention.
- The cluster must self-heal when a failed Pi is brought back online.
- SSH access must be restricted to the local demo network and protected by password-based authentication.

5. TECHNOLOGY STACK

Component	Technology
Infrastructure	5 × Raspberry Pi 4 · microSD card storage · D-Link switch
Web Server	Node.js with Express, orchestrated by K3s with Traefik as reverse proxy and MetalLB for virtual IP
Monitoring	Metrics Server · integrated status page in the website
Dev Tools	SSH · kubectl

6. PROJECT PLAN

Phase	Description	Hours
Phase 1	Planning, hardware assembly and base OS installation on all 5 Pis	**2 h**
Phase 2	Network setup, K3s cluster, Traefik and MetalLB configuration, failover testing	**7 h**
Phase 3	Website deployment with integrated monitoring page and automatic scaling logic	**9 h**
Phase 4	Demo rehearsal, documentation and open-day setup	**7 h**
Total		**25 h**

7. SUCCESS CRITERIA

- Unplugging any single Pi keeps the website reachable, with visible recovery on the monitoring page within 5 seconds.
- As traffic load changes, Traefik visibly redistributes requests across the active Pis and K3s scales the number of containers without any manual intervention.
- The monitoring page clearly shows which Pis are online, their resource usage and which are currently serving traffic.
- A non-technical visitor can understand the demo within one minute without verbal explanation.
- The cluster runs continuously through the full Porte Ouverte without manual intervention.
- Documentation allows another student to reset and redeploy the cluster from scratch.