SysSentry — 7-Day Cloud Deployment & DevOps Plan

■ You now have a clear weekly DevOps + full-stack roadmap for taking your project from a working prototype (SysSentry) → to a production-ready, cloud-deployed observability platform.

■ Quick Recap — Where You Are Now
You already have:
- Working Python backend (SysSentry) → collects metrics, stores in Supabase.
- Streamlit dashboard (local visualization).
- Dockerfile ready for containerization.
- Supabase as your cloud database.

Next, you'll integrate:
- GitHub repo → CI/CD pipeline
- EC2 deployment
- Real-time features (Socket.io or API Gateway WebSockets)
- Log downloads & notifications

■ Weekly Execution Plan

Day 1 (Nov 5) — GitHub Setup & Repo Hardening
Goal: Move your local project (SysSentry) into a professional GitHub repo.

Steps:
1. Create GitHub repo, add .gitignore, LICENSE, README
2. Push local project to GitHub
3. Protect main branch, enable Dependabot

Deliverable: Secure GitHub repo with full project uploaded.

Day 2 (Nov 6) — AWS Setup
Goal: Prepare EC2 instance for deployment.

Steps:
1. Create key pair, security group, IAM role
2. Launch EC2 (Amazon Linux 2 / Ubuntu 22.04)
3. Allow ports 22, 80, 443, 8501

Deliverable: Running EC2 instance accessible via SSH.

Day 3 (Nov 7) — Install Runtime + Deploy Sample
Goal: Run SysSentry on EC2 and confirm connectivity.

Steps:

- SSH into EC2, install Python and Git
- Clone repo, install dependencies, run Streamlit dashboard

Deliverable: SysSentry running live from EC2 (manual start).

Day 4 (Nov 8) — CI/CD Pipeline
Goal: Automate build + deploy via GitHub Actions.

Steps:
- Add workflow file `.github/workflows/deploy.yml`
- Define install, test, and lint stages

Deliverable: CI pipeline runs on every push.

Day 5 (Nov 9) — Docker & Deployment
Goal: Build and deploy SysSentry automatically.

Steps:
- Build and push Docker image to GitHub Container Registry
- Update CI/CD workflow to deploy to EC2

Deliverable: Auto-deploy on main branch push.

Day 6 (Nov 10) — Realtime + Notifications
Goal: Add live streaming and alerting to dashboard.

Steps:
- Use Supabase Realtime / Socket.io
- Add Slack/Email notifications
- Extend dashboard with live refresh

Deliverable: Realtime dashboard + alert system.

Day 7 (Nov 11) — Polishing and Logs
Goal: Add log downloads and finalize for presentation.

Steps:
- Add download button in Streamlit
- Update README with badges and architecture diagram
- Verify Supabase & EC2 integration

Deliverable: Fully deployed monitoring system with logs and dashboard.

■ Architecture Overview
Layer	Purpose	Tools

Application	SysSentry monitoring & alerts	Python, Streamlit, Supabase
Infrastructure	Hosting & monitoring	AWS EC2, CloudWatch
Automation	CI/CD pipeline	GitHub Actions, Docker
Realtime	Instant updates	Socket.io / Supabase Realtime
Security	Secrets, IAM roles	GitHub Secrets, AWS IAM
Observability	Metrics & logs	CloudWatch + Supabase