

DIARIO DI BORDO

Settimana	Obiettivo	Risultati	Componente coinvolta	Riflessioni/Criticità
1	Setup dell'Architettura	Organizzazione del lavoro e divisione dei compiti, configurazione della VM, accesso SSH, installazione del Docker, Firewall.	Ubuntu server e Docker/Docker Compose.	Concentrazione sul setup della persistenza Docker (evitare la perdita di dati) e sull'orchestrazione dei servizi.
2	Raccolta Dati e Monitoraggio	Implementazione dello script monitor.py (Python). Acquisizione e persistenza delle metriche (CPU, RAM, Disco) nel DB ogni X minuti. Logica di	Python 3+ (Core Logic) , monitor.py, Libreria psutil.	Verificare la corretta schedulazione degli script nel container smartdoctor_cron_job. I dati devono fluire correttamente verso PostgreSQL.

		recupero dei valori massimi implementata per l>alerting.		
3	Intelligenza: Autocorrezione e Notifica	Logica di Autocorrezione implementata in alert_check.py. Calcolo della nuova soglia. Invio di alert critico e notifiche di autocorrezione via Telegram.	alert_check.py, Telegram API, config.json	Il trigger dell'autocorrezione deve scattare solo se il superamento è maggiore del 15%. Assicurarsi che config.json sia aggiornato correttamente come float.

4	Finalizzazione e output	Interfaccia Grafana configurata per la visualizzazione delle serie storiche. Test completo del sistema per assicurare la proattività e l'affidabilità. Codice finale e documentazione completati.	Grafana (Interfaccia Utente), PostgreSQL (Query Dati Storici).	Verificare la chiarezza dei messaggi Markdown su Telegram. Confermare l'obiettivo di azzerare l'intervento umano per la configurazione delle soglie.
---	--------------------------------	---	--	--