JM4.1 - Avviare tutto sul proprio PC

In una frase. Sul PC di chi sviluppa ADSWnext si avvia con un solo comando, che fa le domande giuste e accende i servizi scelti.

Per provare una modifica non serve il server di prova. Si accende tutto sul proprio PC, si guarda il risultato e solo dopo si propone la modifica. Il comando di avvio esiste perché i pezzi sono tanti: metterli insieme a mano, ogni volta, è il modo più facile per sbagliare.

1. **Installa Docker.** È l'unica cosa che serve davvero: i servizi girano in contenitori, come sul server di prova.
2. **Copia i file delle impostazioni dai modelli.** Nel codice ci sono solo i modelli, senza valori segreti. Le copie restano sul tuo PC e non entrano mai nel codice. Un valore vuoto nel modello è un segreto o un dato del tuo ambiente: chiedilo a chi segue il progetto.
3. **Lancia il comando di avvio.** Da Windows è `.\shell\run.ps1`. La prima volta ci mette un minuto in più, perché prepara se stesso.
4. **Rispondi alle domande.** Alla prima scegli **Tenant**: è l'installazione completa del gestionale. Poi scegli Docker, l'ambiente di sviluppo, il tipo di database dei Dati aziendali e i servizi da avviare. I servizi di base sono già spuntati.
5. **Salva la configurazione.** Alla fine puoi darle un nome e farla diventare quella predefinita. Dalla volta dopo riparti senza domande.
6. **Apri l'app.** Quando i servizi sono partiti, l'app risponde sul tuo PC alla porta 4200. In sviluppo il controllo di chi entra è spento, se non lo accendi tu.

Quando cambi il codice di un servizio non devi riavviare tutto: ci sono comandi che ricostruiscono un servizio solo.

> **Attenzione.** Il tuo PC non è il server di prova. Qui il database gira in un contenitore con dati tuoi; lì è esterno e condiviso. Una modifica alla forma di un archivio si prova sul database del proprio PC, non su quello condiviso.

> **Per chi sviluppa.**
> • Avvio: `shell/run.ps1` (Windows) o `shell/run.sh`. Ripartenze: `--default`, `--last`, `--last --rebuild --services <servizio>`. Guida del comando: `tools/run-cli/README.md`.
> • Modelli da copiare: `config/.env.dev.example` (la copia si chiama uguale, senza `.example`) e `backend/middleware/sigea-cop-ai/.env.example` nel suo `.env`. Il secondo è obbligatorio anche se non usi quel servizio: senza, lo stack non parte.
> • Ricostruire un servizio solo: `shell/dev-ade-rebuild.ps1`, `shell/dev-sigea-rebuild.ps1`, `shell/dev-gpe-rebuild.ps1`. Come funziona: `devops/docker/README-dev-build.md`.

- La guida d'ingresso completa, con requisiti e ciclo di lavoro, è `docs/getting-started.md`.

Revisione #2
Creato 17 novembre 2026 07:59:38 da Giuseppe Anelli
Aggiornato 17 novembre 2026 09:39:04 da Genesys Software