



id: SI\_001

model: qwen/qwen3.6-plus

seeds:

04\_risiko: sårbarhet

03\_relasjon: konkurranse

05\_handling: implementer

01\_perspektiv: strategisk

02\_tidhorisont: langsiktig

## Instans SI\_001

**Seeds:** strategisk | langsiktig | konkurranse | sårbarhet | implementer

For et helsedata-miljø er swarm ikke tech-smykke, men en strategisk differensiator. Copilot gir individuell ytelse; sverm skaper konkurranseforsprang via parallell innsikt og skalerbar saksbehandling. Men sårbarheten krever null-toleranse for data-lekkasjer.

**Arkitektur & Trade-offs:** Velg **Azure AI Foundry (Prompt Flow) + Azure Container Apps** over Copilot Studio. Studio er dialog-drevet og skal ikke orkestrere 100 samtidige agenter. Eksterne rammeverk (CrewAI, Claude Swarm) bryter med Microsoft-governance og øker sårbarheten ved dataeksport. Azure-native gir Entra ID-kontroll, data residency, og langsiktig eierskap uten vendor-lock-in.

**Agent-spesialisering & Consensus:** Unngå prompt-duplisering. Bruk dynamisk kontekstbinding: `system = f"{GLOBAL_CONTEXT}\nROLLE: {role}\nLINSE: {lens}\n\nOUTPUT_SCHEMA: {schema}"` For Case 1, kjør en **Consensus Arbiter** (én sterk agent) som mottar validert JSON, veier output etter domenekredibilitet, og filtrerer støy. For Case 2, bruk **Azure Service Bus** som innboks. Hver melding trigger en Function som allokterer ledig agent fra pool. Del system-template, men injiser case-spesifikk kontekst via RAG. Consistency sikres via streng JSON-validering før lagring.

**Sikkerhet & Observerbarhet:** Implementer zero-trust: Azure Cognitive Search med Private Endpoints og Managed Identity. Data forblir i tenant; kun tokens/embeddings sirkulerer. Spor alt med OpenTelemetry + Application Insights ( `input_tokens` , `output_tokens` , `error_rate` ). Kostnadskontroll: bruk `gpt-4o-mini` til pre-screening, cache like embeddings, og auto-stop ved 85% budsjett. Batch-timer er optimalt for helsedata.

**Transisjon & Feilmoder:** Tidslinje: Mnd 1-2 (pilot 5 agenter, anonymisert data), Mnd 3-4 (Service Bus + Arbiter), Mnd 5-6 (produksjon). Quick win: parallell journalsegmentering. Sverm feiler ved enkle CRUD-oppgaver, kontekst-overload, eller



manglende schema-validering. Bruk det kun når kompleksitet overstiger lineær kapasitet.

**Neste steg:**

1. Definer Entra ID Managed Identities for alle Azure-komponenter og kartlegg Dataverse-tilganger.
2. Bygg Arbiter-agenten i Azure AI Foundry og test JSON-validering på 10 historiske helsecases.
3. Konfigurer cost-gates med Azure Budgets + auto-stop før pilot-start.