ChatGPT

Safety for Autonomous Agents

1. **OED** – Ranked 1st. All assessments describe the odd-demonstration repo as exceptionally safe for agents (e.g. "**YES — Exceptionally Safe**" with dozens of enforced invariants [1]). It has rigorous fail-closed gates, hermetic builds, and schema validation, making it one of the most agent-safe codebases.
2. **Dapr** – Ranked 2nd. The Dapr repo is also rated extremely safe (Claude Sonnet: "**Verdict: YES – This is an EXEMPLARY environment for autonomous agents**"). It has comprehensive CI and governance (DCO, CODEOWNERS, etc.) that make it agent-ready. OED edges it slightly by implementing more custom safety invariants, but both far exceed GM.
3. **GM (microservices-demo)** – Ranked 3rd. The Google demo is only *moderately* safe ("**Yes, with caveats** — a **moderately safe** environment" [2]). It has CI gates and container isolation, but lacks coverage enforcement and security scanning, and uses mock services, so agents must use caution.

Enterprise-Grade System Readiness

1. **Dapr** – Ranked 1st. Dapr is explicitly enterprise-grade (CNCF-graduated, gold-standard distributed system). Assessments unanimously affirm it as meeting enterprise expectations. For example, Claude Sonnet notes Dapr is the "**definition of enterprise-grade distributed systems platform**" with mature architecture and production-hardened pipelines.
2. **OED** – Ranked 2nd. The odd-demonstration repo is surprisingly production-ready, exceeding typical standards. Claude Sonnet's report says it "**exceeds most enterprise standards**" with FAANG-level safety patterns and a ~1000-line CI pipeline [3]. It even implements full observability and strict testing policies. It falls just short of Dapr only by virtue of being a demo project (vs. Dapr's wide production use).
3. **GM (microservices-demo)** – Ranked 3rd. The Google demo is a well-engineered *reference architecture* but not enterprise-ready. Claude Opus explicitly concludes **"No – … not enterprise-grade"** [4]. It has excellent polyglot patterns and IaC, but critical gaps (minimal tests, no auth, no DR) prevent it from meeting enterprise standards.

Development Effort Required to Build Equivalent

1. **GM (microservices-demo)** – Ranked 1st (least effort). Estimates for replicating the Google demo are smallest. Multiple analyses concur it would take on the order of a few months. For example, Claude Opus estimates **2–4 months** (≈400–800 engineering hours) for an experienced team to recreate the microservices-demo [5].
2. **OED** – Ranked 2nd. Building an odd-demonstration–equivalent is harder. Claude Sonnet estimates about **6–9 months** with a senior team. (One analysis even listed ~42–54 weeks total, though a skilled team might hit 6–9 months.) This reflects its polyglot services, strict invariants, cross-platform CI, etc.
3. **Dapr** – Ranked 3rd (most effort). Recreating Dapr from scratch is orders of magnitude larger. Every assessment notes multi-year effort: e.g. "**3–5 years** with 15–20 person specialized team". Dapr's

scope (service invocation, state management, pub/sub, actors, bindings, etc.) and maturity make it far more costly to replicate than the others.

Each ranking is based solely on the provided assessments. For example, OED's assessment emphasizes its "47 enforced invariants" and hermetic CI [1], while GM's clearly notes missing security and limited tests [6]. Similarly, Dapr's evaluation highlights its CNCF graduation and full feature set, reflecting its high development cost.

---

[1] [3] oed-claude-sonnet-assessment-2025-12-27.md
file://file-GuXoooBcer1w18soPtt2sr

[2] [4] [6] gm-claude-opus-assessment-2025-12-27.md
file://file-1m2zYoiDH5LXUSsXKqESic

[5] gm-claude-sonnet-assessment-2025-12-27.md
file://file-UwbAuFG8zCCjRgBaCu1j87