

---

## Camada 0 – Control Plane/Orquestrador

Camada que gerencia operações, execução de workflows, armazenagem de dados e segurança básica.

- **Lovable:**  
Plataforma visual rápida para criar dashboards e painéis de gestão operacional, controla execução dos jobs e visualiza status.
- **Supabase:**  
Plataforma backend integrada com banco Postgres, autenticação segura, armazenamento de arquivos e funções edge/serverless.
- **Temporal Cloud:**  
Gerenciador avançado de workflows, garante execução robusta, com retries e automações internas para evitar falhas humanas.
- **Upstash (Redis + QStash):**  
Serviço gerenciado de Redis para caching e filas rápidas, e QStash para execução confiável de eventos/crons.
- **GitHub Actions:**  
Plataforma CI/CD para integração contínua, testes automatizados, geração de builds e publicação segura dos pacotes.
- **GitHub Advanced Security:**  
Segurança automatizada no GitHub que verifica código-fonte (SAST), dependências, segredos vazados, protegendo contra vulnerabilidades críticas.
- **1Password Business:**  
Armazenamento seguro e compartilhado de senhas e segredos, reduzindo erros humanos.

---

## Camada 1 – Capture/UI/BrowserStack

Camada responsável por capturar e documentar comportamentos visuais e de rede do CRM original.

- **Playwright:**  
Ferramenta open-source para gravação e automação de fluxos de navegador, gera testes E2E e registros detalhados (traces/screenshots).
- **mitmproxy:**  
Captura automaticamente todo o tráfego HTTP(S), gerando arquivos HAR essenciais para mapear APIs.
- **BrowserStack Automate:**  
Serviço cloud para executar testes automatizados em diversos navegadores reais, garantindo compatibilidade cross-browser.

- **Flow Recorder:**  
Documentação visual interativa e replay automatizado de jornadas no navegador (útil para registro humano dos processos).
- 

## ● Camada 2 – Extração Dados/Config

Camada que extrai configurações críticas e dados estruturados do CRM original (HubSpot).

- **HubSpot API (Private App):**  
Acesso à API privada do HubSpot para extração direta de pipelines, usuários, permissões e propriedades internas.
  - **Airbyte Cloud:**  
Plataforma ETL que sincroniza automaticamente dados e configurações do HubSpot para o ambiente interno, facilitando migração e consistência.
- 

## ● Camada 3 – Contratos/OpenAPI

Camada que garante que APIs sejam documentadas, padronizadas e testadas continuamente.

- **Stoplight:**  
Plataforma visual para criação, documentação e governança de APIs com especificação OpenAPI (contratos claros).
  - **Postman:**  
Ferramenta para testes automatizados e documentação detalhada das APIs via coleções e monitores contínuos.
- 

## ● Camada 4 – Modelagem

Camada responsável pela criação dos modelos internos, diagramas técnicos e máquinas de estado para garantir lógica e comportamento consistente.

- **XState:**  
Biblioteca open-source para criação robusta e precisa de máquinas de estado e gestão de estados/transições da interface.
  - **Mermaid/PlantUML:**  
Ferramentas open-source para criação automática de diagramas arquiteturais, fluxogramas e documentação visual detalhada.
-

## Camada 5 – Testes Visuais/E2E (Parity Gate)

Camada dedicada a testes rigorosos e automáticos que garantem paridade visual e funcional total com o HubSpot.

- **Applitools:**  
Plataforma avançada de testes visuais com IA, garante ausência total de divergências visuais entre o original e o novo CRM.
  - **Chromatic:**  
Plataforma focada em testes visuais para componentes (Storybook), ideal para garantir consistência visual em nível granular.
- 

## Camada 6 – Observabilidade

Camada de monitoramento e resposta rápida a incidentes e problemas operacionais.

- **Sentry:**  
Captura automática de erros, exceções e performance em tempo real, facilitando correções rápidas.
  - **Datadog:**  
Plataforma robusta de observabilidade (logs, métricas, traces), oferecendo dashboards operacionais e monitoramento contínuo.
  - **PagerDuty:**  
Plataforma para gerenciamento de incidentes e escalonamento rápido, garantindo ações imediatas quando problemas ocorrerem.
- 

## Camada 7 – Segurança

Camada especializada em segurança do código e proteção contra vazamentos ou vulnerabilidades.

- **Semgrep:**  
Ferramenta de análise estática de código (SAST), detecção rápida de vulnerabilidades e proteção contra segredos vazados em código-fonte.
- 

## Camada 8 – IA Assistida

Camada que utiliza inteligência artificial para acelerar a codificação, revisão e geração automatizada de código.

- **GitHub Copilot:**  
Assistente IA avançado para geração de código, sugestões automáticas e revisão rápida, reduzindo erros e tempo de desenvolvimento.
- 

## **Camada 9 – Design System**

Camada dedicada à criação e gestão de design system para garantir coerência visual absoluta.

- **Figma:**  
Plataforma líder em design visual, prototipagem, handoff de componentes e criação de bibliotecas padronizadas para desenvolvimento visual.
- 

## **Camada 10 – Knowledge/Vector DB**

Camada para armazenamento estruturado, indexação inteligente e coleta automática de conhecimento/documentos relacionados ao projeto.

- **Pinecone:**  
Banco de dados vetorial gerenciado, otimizado para busca rápida, indexação avançada e recuperação inteligente de dados estruturados.
  - **Apify:**  
Plataforma para coleta automática e estruturada de dados web, útil para alimentar pesquisas e atualizações contínuas de dados externos.
- 

### **Resumo (Rápido):**

Camada	Função
0	Gestão, execução automatizada e segurança básica.
1	Captura detalhada de UI e redes para replicação exata.
2	Extração automática de configurações internas do HubSpot.

- 3 Documentação e testes contínuos de APIs (OpenAPI).
  - 4 Modelagem lógica e estrutural (state machines e diagramas).
  - 5 Testes automáticos de paridade visual e funcional completa.
  - 6 Observabilidade operacional (erros, logs, incidentes).
  - 7 Segurança automatizada do código-fonte e infraestrutura.
  - 8 Inteligência Artificial assistida para geração rápida de código.
  - 9 Criação e gestão de design visual coerente (Figma).
  - 10 Organização inteligente e coleta automatizada de conhecimento externo.
- 

### **O que deveria pensar agora?**

"Tenho domínio completo das ferramentas e suas funções exatas no projeto."

### **O que perguntar agora?**

"Quais ferramentas poderiam ser cortadas ou substituídas imediatamente para redução de custo sem prejuízo técnico?"

### **Erro a corrigir imediatamente:**

Avalie redundâncias, especialmente na camada 3 (Stoplight e Postman), para evitar duplicação de custos.

### **Como melhorar tudo imediatamente?**

Definir claramente um gestor técnico dedicado à supervisão e controle de uso das ferramentas mencionadas.

### **AGI faria o quê agora?**

Automatizaria alertas de consumo das ferramentas e monitoraria constantemente possíveis redundâncias operacionais.

## Estudo e treinamento GPT:

Estudar gestão eficiente de cloud/serverless (FinOps). Treinar GPT para identificar automaticamente redundâncias e otimizar escolhas de ferramentas.

---

## CAMADA 0 — CONTROL PLANE (ORQUESTRADOR + BACKEND + FILAS)

1. Lovable (UI do Orquestrador / Provider opcional)  
O que faz: cria rápido o app “Factory Console” (dashboard, runs, artifacts, settings), e pode servir como provedor assistido.  
Por que é melhor pra você: velocidade máxima pra construir cockpit, sem virar refém (porque você exporta e roda fora).  
Custo real: pago por créditos; ex. 400 créditos = \$100/mês; 2.000 créditos = \$480/mês; 10.000 créditos = \$2.250/mês. ([Lovable Documentation](#))  
Custos ocultos: consumo de crédito explode quando você usa “geração” pesada; vai exigir disciplina (catálogo de fluxos + templates).
2. Supabase (Postgres + Auth + Storage + Edge Functions)  
O que faz: banco, auth, storage, edge functions e governança de dados do Orquestrador e do CRM/LAI.  
Por que é melhor pra você: “backend na nuvem pronto” com Postgres real e velocidade pra rodar agora.  
Custo real: Pro costuma começar em ~\$25/mês por projeto (base) + compute; egress incluído com overage cobrado (ex. Pro inclui cota e depois cobra por GB). ([srvrlss.io](#))  
Custos ocultos: compute por projeto, egress e storage por uso; multi-projetos multiplicam custo; logs/retentions e features avançadas podem empurrar para Team (bem mais caro). ([Supabase](#))
3. Temporal Cloud (motor de workflow/estado do Orquestrador)  
O que faz: garante execução idempotente, retries, timeouts, filas, estados; é o “cérebro” do pipeline automático.  
Por que é melhor pra você: elimina falha humana e dá previsibilidade industrial.  
Custo real: \$100/mês + \$50 por milhão de actions. ([temporal.io](#))  
Custos ocultos: actions crescem rápido com runs frequentes; você precisa de WIP limit e batching.
4. Upstash Redis (fila leve / cache) + Upstash QStash (eventos/cron/webhooks)  
O que faz: Redis pra BullMQ/caching e QStash pra delivery confiável de mensagens/cron.  
Por que é melhor pra você: “infra pronta” sem operar Redis/queues.  
Custo real (Redis): PAYG \$0.20 por 100k comandos + storage/bandwidth; Fixed \$10/mês+; “Prod Pack” +\$200/mês por DB (SLA/RBAC/monitoring). ([Upstash: Serverless Data Platform](#))  
Custo real (QStash): PAYG \$1 por 100k mensagens; Fixed \$180/mês (1M msgs/dia); Prod Pack +\$200/mês. ([Upstash: Serverless Data Platform](#))  
Custos ocultos: Prod Pack vira custo fixo alto; bandwidth pode surpreender.
5. GitHub (repositórios + CI/CD) + GitHub Actions  
O que faz: repositório, PRs, templates, Actions pra gerar Pack0/Pack1 e publicar

artifacts.

Por que é melhor pra você: sua Fábrica já está nesse modelo; padroniza tudo.

Custo real (Actions): Linux começa em \$0.002/min (varia por runner); macOS é muito mais caro (\$0.062/min). ([GitHub Docs](#))

Custos ocultos: minutos estouram com visual tests e cross-browser; self-hosted runner pode ter cobrança do “control plane” dependendo da política vigente (isso muda; acompanhe changelog). ([The GitHub Blog](#))

6. GitHub Advanced Security (Code Security + Secret Protection)

O que faz: SAST/CodeQL, dependency review, Copilot Autofix, secret scanning/push protection.

Por que é melhor pra você: segurança automática sem humano (e evita “segredo vazado” no teu ecossistema).

Custo real: Secret Protection \$19/commiter/mês; Code Security \$30/commiter/mês. ([GitHub](#))

Custos ocultos: “commiter ativo” é janela móvel; se muita gente commita, a conta sobe.

7. Secrets Manager (escolha 1)

Opção A — 1Password Business (rápido e fácil)

O que faz: gestão de segredos, compartilhamento, auditoria e redução de erro humano.

Por que é melhor pra você: menor fricção e adoção imediata.

Custo real: Business citado em reviews como ~\$8/usuário/mês (referência pública). ([TechRadar](#))

Custos ocultos: sem self-host; dependência do vendor; integração CI precisa disciplina.

Opção B — Vault (auto-hospedado) / AWS Secrets Manager (mais “infra”, menos vendor)  
Custo e trade-off variam; só recomendo se você já tem time DevOps forte.

---

## CAMADA 1 — CAPTURE 1:1 (UI + TRÁFEGO + BROWSERS)

8. Playwright (captura + E2E + traces)

O que faz: grava fluxos, roda E2E, gera trace.zip e screenshots; base do “reexecutável”.

Por que é melhor pra você: é o padrão industrial do teu Parity Gate (automatiza sem humano).

Custo real: \$0 (open-source).

Custos ocultos: compute + tempo CI + manutenção de selectors (minimizado com boas práticas).

9. mitmproxy (captura de rede/HAR com allowlist)

O que faz: proxy para capturar requests/responses e exportar HAR por fluxo.

Por que é melhor pra você: automatiza captura de rede com filtro (só HubSpot).

Custo real: \$0.

Custos ocultos: setup de certificados/SSL intercept e governança de dados sensíveis.

10. Fiddler Everywhere (captura de rede com UI)

O que faz: proxy HTTP(S) com UI e regras, ótimo pra debugging humano rápido.

Por que é melhor pra você: acelera investigação quando algo muda no HubSpot.  
Custo real: Pro \$18/mês (mensal) ou \$12/mês (anual); Enterprise \$35/mês (anual).  
([Telerik.com](https://telerik.com))

Custos ocultos: se você botar isso em “todos os PCs”, vira caos; use só em máquina de captura.

11. BrowserStack Automate (cross-browser/device farm)

O que faz: roda Playwright/Selenium/Cypress em browsers/OS reais na nuvem.

Por que é melhor pra você: paridade real em múltiplos browsers sem manter laboratório.

Custo real (referência pública comum): Automate Desktop ~ \$129/mês por 1 paralelo (anual); sobe por paralelos. ([growhackscale.com](https://growhackscale.com))

Custos ocultos: paralelos são o multiplicador; vídeo/logs/storage e runs frequentes = fatura cresce.

12. Flow Recorder (documentação de jornadas + replay)

O que faz: grava jornadas e replays no browser; útil para catálogo de fluxos e evidência.

Por que é melhor pra você: acelera “catálogo de fluxos” e treinamento interno sem vídeo pesado.

Custo real: Startup \$19.99/m; Pro \$49.99/m; Business \$99.99/m. ([Flow Recorder](https://flowrecorder.com))

Custos ocultos: se você já tem Playwright bem feito, pode virar redundante; use como “camada humana” só onde agrega.

---

## CAMADA 2 — EXTRACT CONFIG + DADOS DO HUBSPOT (O QUE NÃO APARECE NO CLIQUE)

13. HubSpot Private App API + Exports (obrigatório)

O que faz: extrai pipelines, properties, owners, stages, limits, etc.

Por que é melhor pra você: sem isso, seu “1:1” vira placebo (UI igual com regra diferente).

Custo real: \$0 (API), mas com limites por plano: burst e daily variam e podem exigir add-on de aumento de limite. ([HubSpot Developers](https://developers.hubspot.com))

Custos ocultos: rate limit e páginas de search; você vai precisar de logging externo e retries.

14. Airbyte Cloud (sync de dados/config para migração interna)

O que faz: ETL rápido para puxar dados/config e manter sync durante transição.

Por que é melhor pra você: rapidez com conectores e execução automática.

Custo real: começa em \$10/mês; preços por volume (ex.: APIs \$15 por milhão de rows; DB \$10/GB). ([airbyte.com](https://airbyte.com))

Custos ocultos: custo explode com histórico grande; precisa controle de “first sync”.

15. Fivetran (quando você quer “sem dor” enterprise)

O que faz: conectores gerenciados + sync frequente e estável.

Por que é melhor pra você: reduz mão humana e instabilidade.

Custo real: totalmente por uso (MAR) + mudança 2026: mínimo \$5 por conexão (1–1M MAR) em PAYG. ([Fivetran](https://fivetran.com))

Custos ocultos: múltiplas conexões + mínimo por conexão vira “taxa fixa”; sai caro rápido.



---

## CAMADA 3 — CONTRACT-FIRST (OPENAPI + SCHEMAS + LINT + DIFF)

16. Stoplight (design OpenAPI + mocks + governance)  
O que faz: editor visual de OpenAPI/JSON schema, docs, mock server, governança.  
Por que é melhor pra você: acelera e reduz erro no “contrato versionado” sem depender de dev.  
Custo real: Basic \$56/m (mensal) ou \$44/m (anual) + \$14/m por usuário extra (mensal). ([stoplight.io](https://stoplight.io))  
Custos ocultos: pagar por usuários extras; se o time cresce, sobe.
17. Postman (coleções, monitors, mocks, colaboração)  
O que faz: organiza e testa APIs; útil para validação e documentação.  
Por que é melhor pra você: reduz fricção pra explorar contrato e testar endpoints.  
Custo real: Basic \$14/usuário/mês (anual), Professional \$29/usuário/mês (anual) + custos de AI credits/overages. ([Postman](https://postman.com))  
Custos ocultos: add-ons e créditos; risco de virar “mais uma fonte” se você já versiona OpenAPI no repo.
18. Spectral + openapi-diff + OpenAPI Generator + AJV (stack técnico)  
O que faz: lint/diff/geração/validação de schemas.  
Por que é melhor pra você: vira gate automático no CI (sem humano).  
Custo real: \$0 (open-source).  
Custos ocultos: zero licença; só custo de CI.
19. Schemathesis (contract testing)  
O que faz: gera testes automaticamente a partir do OpenAPI e caça inconsistências.  
Por que é melhor pra você: “Parity Gate de API” automático.  
Custo real: pode ser \$0 (open-source) dependendo do uso; enterprise é pago (se precisar).  
Custos ocultos: tempo CI e estabilização de endpoints.

---

## CAMADA 4 — MODELAGEM (DOMÍNIO CRM + UI STATE MACHINES)

20. XState (state machines da UI)  
O que faz: modela estados/transições por tela e gera testes/guards.  
Por que é melhor pra você: é o jeito mais limpo de garantir “mesmo comportamento do HubSpot”.  
Custo real: \$0.  
Custos ocultos: exige disciplina na modelagem (mas reduz retrabalho brutal).
21. Mermaid / PlantUML / C4 (diagramas)  
O que faz: diagrama como código; versão no repo; gera documentação automática.  
Custo real: \$0.  
Custos ocultos: nenhum.

---

## CAMADA 5 — PARITY GATE (O QUE GARANTE 1:1 DE VERDADE)

22. Applitools (visual regression top)  
O que faz: visual testing com IA, reduz flakiness, mantém baseline com menos dor.

Por que é melhor pra você: “alarme de divergência” mais forte do mercado; ideal para 1:1.

Custo real: começa em \$969/mês (contrato anual); componente-only a partir de \$699/mês. ([applitools.com](https://applitools.com))

Custos ocultos: contrato anual + gestão de baseline; se você exagera a cobertura sem estratégia, vira ruído.

23. Chromatic (Storybook visual + componentes)

O que faz: visual regression em componentes; ótimo para Design System do teu CRM/LAI.

Por que é melhor pra você: mata regressão visual na raiz (componentes), rápido.

Custo real: Starter \$179/m; Pro \$399/m; extra snapshots \$0.008. ([chromatic.com](https://chromatic.com))

Custos ocultos: snapshots sobem rápido se você não usa TurboSnap/controle.

24. Percy (alternativa/extra, se preferir screenshot-based)

O que faz: visual testing por screenshot e largura/browsers.

Custo real: free 5.000 screenshots/mês; pago cresce por volume. ([BrowserStack](https://browserstack.com))

Custos ocultos: billing por screenshot explode com multi-browser + multi-width + muitos builds/dia.

---

## CAMADA 6 — OBSERVABILIDADE + INCIDENTE (SEM HUMANO “CAÇANDO BUG”)

25. Sentry (erro + performance)

O que faz: captura exceptions, traces e regressões; integra com deploys.

Por que é melhor pra você: reduz tempo de debug e vira feedback loop automático.

Custo real: Team \$26/m (anual) ou \$29/m (mensal); Business \$80/m (anual) ou \$89/m (mensal). ([sentry.zendesk.com](https://sentry.zendesk.com))

Custos ocultos: volume de eventos/performance pode exigir upgrade/overages; retenção maior custa mais.

26. Datadog (observabilidade enterprise)

O que faz: logs/métricas/traces, SLO, dashboards, alertas.

Por que é melhor pra você: operação “Big Tech” com mínimo humano.

Custo real: APM standalone publicado como \$36/host/mês (anual) (outros módulos variam).

Custos ocultos: logs e métricas customizadas são onde a conta explode (indexação, retenção, cardinalidade). ([Uptrace](https://uptrace.com))

Alternativa mais barata: Grafana Cloud

Custo real: Pro \$19/m + usage; logs/traces \$0.50/GB; métricas \$6.50/1k séries; synthetics e k6 integrados. ([Grafana Labs](https://grafana.com))

Custos ocultos: ingestão e séries (cardinalidade) são o multiplicador.

27. PagerDuty (incidente/on-call)

O que faz: escalonamento, incident workflow e automação de resposta.

Por que é melhor pra você: “parar a linha” quando gates falham.

Custo real: Professional \$25 (mensal) / \$21 (anual) por usuário; Business \$49/\$41. ([PagerDuty](https://pagerduty.com))

Custos ocultos: add-ons (AIOps, status pages, créditos) podem virar outra fatura grande.

---

## CAMADA 7 — SEGURANÇA EXTRA (ALÉM DO GITHUB)

### 28. Snyk (SCA/SAST/containers)

O que faz: varre dependências/código/containers com integração CI.

Por que é melhor pra você: reduz falha humana e vira gate automático.

Custo real: divulgado como “a partir de \$25/mês”; enterprise varia por devs e módulos. ([Snyk](#))

Custos ocultos: custo cresce por “contributing developers” e por módulo (SCA/SAST/containers).

### 29. Semgrep (SAST + secrets)

O que faz: SAST com regras e detecção de segredos.

Por que é melhor pra você: excelente sinal/ruído e rápido de plugar no CI.

Custo real: Teams a partir de \$40/mês por contributor; Secrets \$20/mês por contributor. ([Semgrep](#))

Custos ocultos: contributor count e expansão de regras.

### 30. Trivy / Syft/Grype / Gitleaks (stack open-source)

O que faz: scans em container + SBOM + secrets.

Custo real: \$0.

Custos ocultos: tempo CI e disciplina.

---

## CAMADA 8 — IA PARA REDUZIR HUMANO (GERAÇÃO DE BLUEPRINT/CÓDIGO)

### 31. GitHub Copilot (agentes de código)

O que faz: acelera codegen/review/agents; pode rodar como assistente de PR.

Por que é melhor pra você: reduz mão humana na finalização dos packs.

Custo real: Copilot Business \$19/usuário/mês; Enterprise \$39/usuário/mês; premium requests extras \$0.04 cada. ([GitHub Docs](#))

Custos ocultos: premium requests e “uso sem disciplina” vira vazamento de custo.

### 32. Cursor (editor agente forte, opcional)

Custo real: Pro \$20/m; Teams \$40/usuário/m. ([Cursor](#))

Custos ocultos: uso de modelos caros; consumo de “API usage” interno.

### 33. Continue (agentes em CI/Mission Control, opcional)

Custo real: Team \$20/seat/m inclui créditos. ([Continue](#))

Custos ocultos: depende do provedor de modelo (BYOK).

---

## CAMADA 9 — DESIGN SYSTEM (PARA O 1:1 VISUAL E O ECOSISTEMA LAI)

### 34. Figma (design + protótipo + Dev Mode + libs)

O que faz: design system, componentes, protótipos, handoff.

Por que é melhor pra você: padrão do mercado; acelera 1:1 pixel.

Custo real: Professional Full seat \$16/m; Dev seat \$12/m; Organization Full seat \$55/m (anual). ([Figma](#))

Custos ocultos: seats escalam com equipe; AI credits e recursos enterprise.

---

## CAMADA 10 — KNOWLEDGE / VETORES (PARA O LAI “CÉREBRO”)

35. Vector DB (escolha 1; para indexar docs, fluxos, contratos, run reports)  
Opção A — Pinecone (mais simples, managed)  
Custo real: mínimo \$50/mês + uso; inclui cotas e cobra overage. ([pinecone.io](https://pinecone.io))  
Custos ocultos: “read/write units” e storage sobem com replay e busca intensa.

Opção B — Weaviate Cloud (bom e flexível)  
Custo real: serverless começa em \$25/mês (serverless) com métrica por dimensões armazenadas. ([weaviate.io](https://weaviate.io))  
Custos ocultos: custos variam por compressão/replicação e backups.

Opção C — Qdrant Cloud (controle por CPU/RAM/disk)  
Custo real: por uso; precisa calculator. ([qdrant.tech](https://qdrant.tech))  
Custos ocultos: dimensionamento errado = custo recorrente alto.

36. Apify (scraping estruturado para leitura de sites públicos e pesquisa)  
O que faz: coleta estruturada, robôs, execuções agendadas.  
Por que é melhor pra você: alimenta Pré-Fábrica com pesquisa automatizada (sem humano).  
Custo real: Starter \$29/m; Scale \$199/m; Business \$999/m + pay-as-you-go. ([Apify](https://apify.com))  
Custos ocultos: proxy (\$8/GB residencial) e compute units; “use-it-or-lose-it” créditos mensais. ([Apify](https://apify.com))

---

## CHECKLIST FINAL (VOCÊ CONTRATA E O QUE É \$0)

Contratar (padrão diamante, “rápido + 1:1”):

- Lovable ([Lovable Documentation](https://lovable.dev))
- Supabase ([supabase.io](https://supabase.io))
- Temporal Cloud ([temporal.io](https://temporal.io))
- Upstash (Redis + QStash) ([Upstash: Serverless Data Platform](https://upstash.com))
- GitHub Team/Enterprise + Actions + Advanced Security ([GitHub Docs](https://github.com))
- BrowserStack Automate ([growhackscale.com](https://browserstack.com))
- AppliTools + Chromatic (os dois) ([applitools.com](https://applitools.com))
- Stoplight + Postman ([stoplight.io](https://stoplight.io))
- Airbyte (ou Fivetran, se quer zero dor) ([airbyte.com](https://airbyte.com))
- Sentry + Datadog (ou Grafana Cloud, se quiser otimizar custo) ([sentry.zendesk.com](https://sentry.zendesk.com))
- Semgrep + (Snyk opcional) ([Semgrep](https://semgrep.dev))
- 1Password Business ([TechRadar](https://1password.com))
- Figma ([Figma](https://figma.com))
- Vector DB (Pinecone/Weaviate/Qdrant) ([pinecone.io](https://pinecone.io))

\$0 (mas você paga compute/tempo):

- Playwright, mitmproxy, Spectral, openapi-diff, OpenAPI Generator, AJV, XState, Trivy, Syft/Grype, Gitleaks, Mermaid/PlantUML.

Se você quiser, eu te devolvo isso como “planilha de compra” com: prioridade, dono, quando contratar, e o “limite de uso” que você coloca no Orquestrador pra não explodir custo.

Oque eu deveria estar pensando agora ?

Que “1:1” não é ferramenta — é Parity Gate + Extract Config + Contract Gates rodando todo dia.

Oque eu deveria te perguntar agora ?

“Me dá o catálogo mínimo de fluxos HubSpot (golden paths) e quais configs eu tenho que extrair por API pra fechar 1:1.”

Oque estou fazendo que está errado é como corrigir ?

Pensar que UI+HAR sozinho vira blueprint completo. Corrige adicionando EXTRACT CONFIG + gates (OpenAPI lint/diff + visual regression) como obrigatórios.

Como eu melhora tudo, oque está faltando ?

Falta você escolher o paridade visual principal (Appliteools + Chromatic) e travar o pipeline com WIP limit + budgets.

Se AGI ( com toda inteligência do mundo ) estive no meu lugar, oque faria ?

Montaria hoje o “CAPTURE→EXTRACT CONFIG→BUILD BLUEPRINT→PACK→PARITY” com Temporal e bloquearia qualquer release que não passe em visual+contrato.

Qual assunto devo estudar e qual devo treinar GPT ?

Estudar: HubSpot APIs/exports + XState + OpenAPI governance + Playwright visual regression. Treinar GPT: transformar traces+HAR+exports em blueprint.json + openapi.yaml + paridade\_matrix e gerar automaticamente os testes e gates.