

Formação de AI Engineering

Douglas Macedo Sgrott

Proposta de desenvolvimento

O presente documento visa servir como documentação de um projeto de Engenharia de IA visando a conclusão do curso de Formação em AI Engineering pela Indiciium.

Segundo o enunciado, que é propositalmente vago, com o objetivo de fomentar diferentes formas de desenvolvimento do trabalho, o objetivo final é a criação de uma solução usando LLM que consiga gerar um relatório com informações, métricas e gráficos relacionados a evolução de SARS no Brasil em um determinado período de tempo.

De tal forma, apesar de não haver um caminho que possa ser considerado como mais correto ou incorreto para o desenvolvimento de tal solução, é possível interpretar pelo enunciado que a criação de um agente que faça uso de ferramentas (tools) seja o resultado desejado.

Este trabalho, no entanto, foi feito com a principal proposta de reproduzir de forma simplificada a estrutura utilizada em um projeto real, como forma de abarcar o desafio e um projeto real, tanto como forma de estudo como de desenvolvimento. Abaixo, segue um resumo com os principais pontos abordados no desenvolvimento:

- **Estrutura:** Foi criado um grafo, com três diferentes ramificações em paralelo (para métricas, gráficos e notícias), que sempre são executados.
- **Agência x Determinismo:** Foi visado gerar um equilíbrio entre capacidades determinísticas e agênticas, que será discutido mais adiante.
- **Tools x Nodos:** Parte das capacidades da solução, consequentemente, não se encontram em tools, e sim nos nodos que compõem o grafo.
- **Guardrails:** Na solução final, o primeiro nodo é um agente que visa determinar a intenção do usuário a partir do prompt. Com isso, é possível não gerar o relatório com as informações necessárias caso o prompt seja off-topic, além de gerar relatórios parciais (apenas com métricas, gráficos ou notícias, ou diferentes combinações).
- **Short Circuit:** Caso o trabalho de um nodo não seja necessário (devido ao guardrail ser ativado por um prompt off-topic ou por um relatório parcial), o nodo é "curto-circuitado", ou seja, é executado mas não gera output relevante.
- **Observabilidade e Governança:** Foram utilizadas 2 frameworks voltados a observabilidade e governança: Langsmith e Langfuse, que serão tratados em uma seção específica.
- **Relatório:** Foi decidido gerar um relatório HTML usando um template previamente construído. Dessa forma, as informações geradas pela solução proposta alimentam o template já criado.

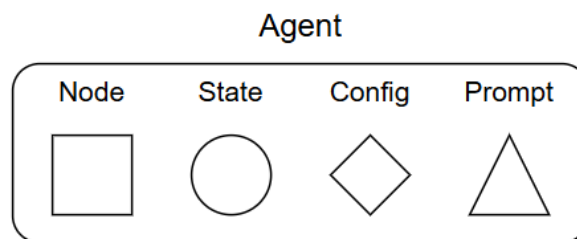
Organização da solução

A estrutura da solução (e do código) reflete de forma simplificada a estrutura vista em um projeto real. Desta estrutura, o que é relevante para este projeto é a definição e organização dos agentes e workloads em suas pastas, ilustradas abaixo:

- /workflows
 - /agents
 - /node
 - /prompts
 - /config
 - /state
 - /shared

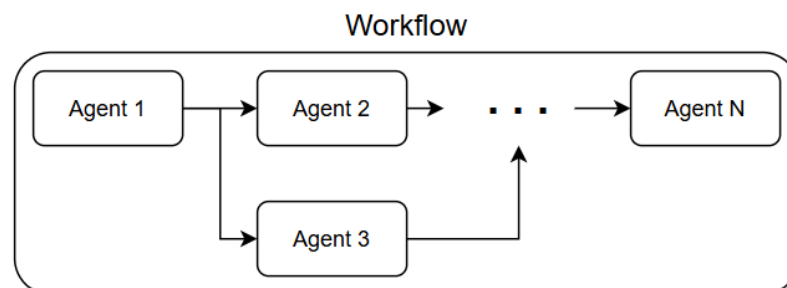
Definição de agente

Um agente pode ser considerado como um nodo no grafo (mas nem todo nodo é um agente), podendo conter a definição do nodo em si, um estado, uma configuração e um prompt. Note que nem todos estes itens são obrigatórios, por exemplo, um agente que não faz uso de LLM não necessita de um arquivo contendo o prompt.

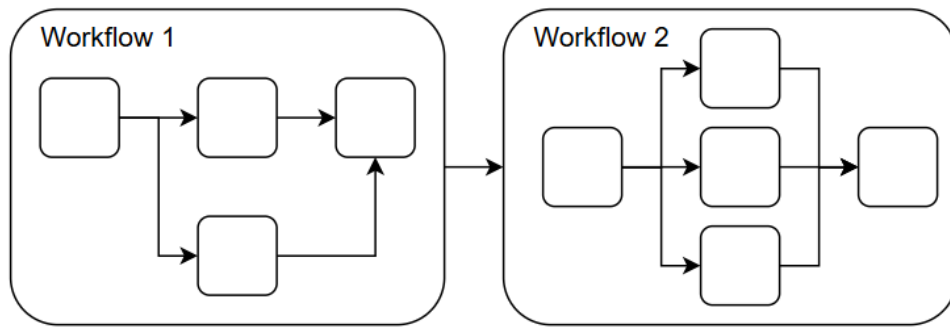


Definição de workload

Um workload é basicamente um grafo ou subgrafo composto por nodos e agentes. Neste trabalho, workload e grafo irão se referir a mesma estrutura.



Um workload pode ser também composto por múltiplos workloads, apesar dessa estrutura não estar presente neste trabalho:



Para a utilização de um workload, é possível criar um script que o instancie e o execute, desacoplando sua definição de sua execução.

Vantagens desta abordagem

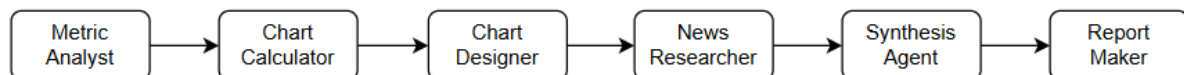
Por focar em modularidade, em tese, seria possível a reutilização de nodos/agentes e workloads na medida que o projeto cresce. Além disso, a divisão de diferentes componentes em diferentes arquivos visa simplificar a manutenção do código.

Estrutura dos workloads

Como citado anteriormente, a estrutura é um grafo onde cada nodo realiza uma determinada tarefa. Ao longo do desenvolvimento, diferentes workloads foram criados para solucionar o problema.

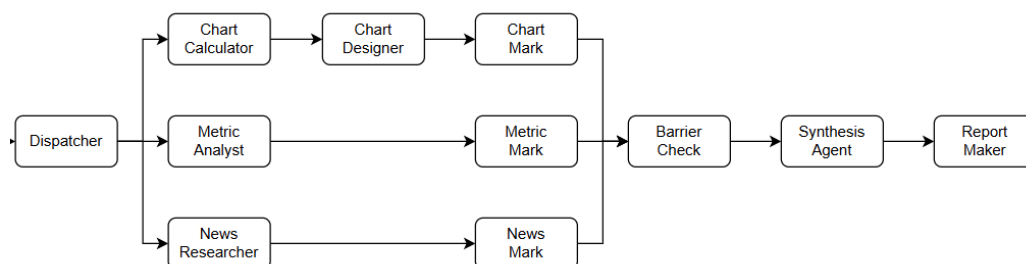
Abordagem simplificada

De forma **simplificada**, Inicialmente a solução foi feita de forma linear, onde todas as tarefas são encadeadas, conforme ilustra figura abaixo:



Abordagem paralela

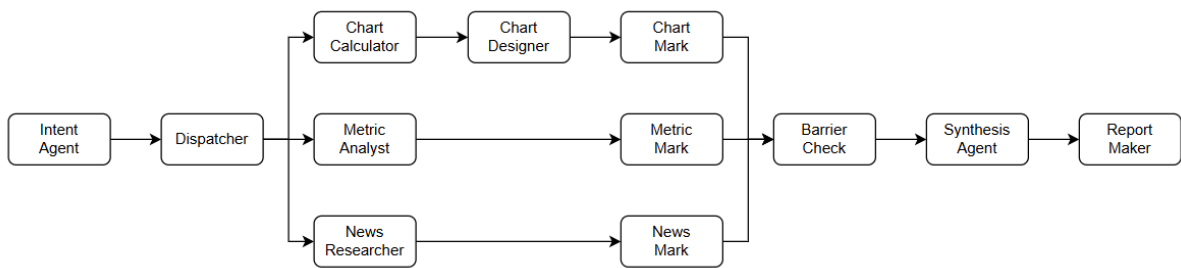
Logo depois, visando uma solução mais **paralela**, como diferentes partes deste gráfico podem operar de forma individual e independente, foi decidido abordar o paralelismo e adicionar o nodo inicial de guardrail:



Devido a condição de corrida (ou race condition) que vem acompanhado de soluções paralelas e assíncronas, foi necessário adicionar nodos para sinalizar se os nodos terminaram seus serviços (Mark and Barrier Check), assim como um nodo responsável em despachar o serviço para todos as ramificações (Dispatcher).

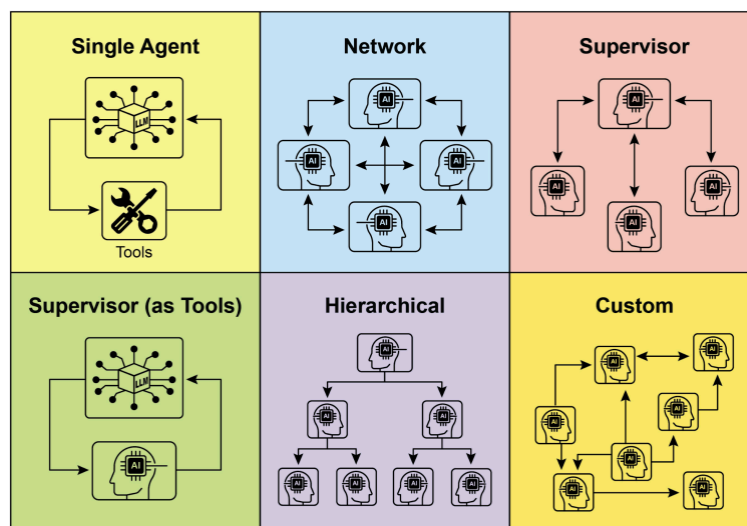
Abordagem final (paralela com guardrails)

Por fim, foi adicionado um nodo para fazer a guardrail do projeto no início do workload, o Intent Agent. Este nodo verifica a intenção do usuário e que tipo de relatório deve ser criado (completo, parcial ou off-topic).



Alternativas

Para fins de completude, e como mencionado anteriormente, uma alternativa a solução proposta seria a utilização de arquiteturas mais agênticas que fazem mais uso de ferramentas/MCP, como ilustrado na imagem abaixo retirada do livro *Agentic Design Patterns*, de Antonio Gulli:



Funções determinísticas e agênticas

O ponto forte de um trabalho que envolve o uso de LLMs pode ser considerado a alta capacidade de adaptabilidade, que vem da imensa capacidade da LLM de realizar trabalhos complexos e entender pedidos elaborados sem a necessidade de extensa produção de código.

No entanto, tamanha adaptabilidade, quando não gerenciada de forma adequada, pode gerar bugs ou falhas na execução de uma tarefa, devido a diferentes fatores, como a complexidade de tarefas interligadas, estocasticidade no comportamento de uma LLM ou variações de saída quando ocorre uma troca de modelo. Isto aumenta ainda mais quando é utilizado uma abordagem agêntica, onde uma LLM determina não somente a forma que algo é feito, mas a sua ordem.

Em um cenário onde o funcionamento da solução é crítica, é importante mapear quais funções são agênticas e quais são determinísticas. Note que aqui, agêntico se refere a capacidade da LLM de criar um resultado de forma autônoma, e determinístico se refere a um resultado que é logicamente programável. A tabela abaixo busca resumir o nível de determinismo e agência de cada agente.

Agente	Aspecto determinístico	Aspecto agêntico
Chart calculator	Baixo, pois apenas o tipo e período da consulta determinado pelo prompt. A saída também é determinada (mas não forçada) pelo prompt.	Médio, pois a consulta SQL é feita pelo agente, mas com informações auxiliares.
Chart designer	Mínimo (pedido de uso de Plotly), pois se adapta de acordo com a saída do Chart calculator	Alto, pois o gráfico é feito inteiramente pelo agente.
Intent agent	Mínimo, pois a saída é 100% definida pelo conteúdo do LLM.	Alto, pois apenas um LLM consegue lidar com textos não estruturados e que podem ter intenções e/ou formulações complexas.
Metric analyst	Médio, pois as colunas a serem usadas, as métricas e suas equações são determinado pelo prompt e dicionário de dados.	Médio, pois a consulta SQL é feita pelo agente, mas com informações auxiliares.
News researcher	Mínimo, pois é necessário capacidades agênticas para buscar informações na web.	Alto, pois se utiliza Tavily agent ou DuckDuckGo agent.
Report maker	Alto, pois o template HTML é previamente construído com Jinja	Baixo, pois o nodo apenas coleta diferentes saídas para compor o template de relatório

Synthesis agent	Mínimo, pois o uso de LLMs é ideal para sínteses complexas.	Alto
-----------------	---	------

É importante salientar que, apesar de alguns nodos (ou suas combinações) terem, em tese, alta capacidade agêntica, a principal fonte de determinismo na solução proposta está na estrutura do grafo em si.

Alternativas

Tal mapa pode ser considerado como importante, pois em um projeto realístico, se faria fundamental entender quais partes podem ser feitas de forma agêntica ou determinística, para entender limitações do projeto, custos, possíveis fontes de bugs, entre outros.

Nesta subseção, é **exemplificado** como alguns agentes poderiam ser de forma extremas, agênticos ou determinísticos:

Cálculo de métricas

Agêntico: a escolha das métricas a serem computadas são decididas pelo agente em si ou informadas pelo usuário através de um prompt. A formulação das métricas é realizada pelo agente, e primeiramente é feita uma consulta ao banco de dados para identificar quais colunas estão presentes e podem ser usadas.

Misto: a consulta SQL e cálculo é feita pelo agente, mas com informações previamente e manualmente definidas.

Determinístico: um script (Python/SQL/Spark) acessa programaticamente a fonte de dados com filtros específicos, seleciona colunas previamente definidas, e realiza o cálculo das métricas com funções previamente estabelecidas.

Criação de gráficos

Agêntico: a escolha dos gráficos a serem criados é decidido pelo agente em si ou informado pelo usuário através de um prompt. Colunas a serem utilizadas, consulta SQL, criação da função que gera o gráfico, assim como tipo de gráfico, biblioteca a ser utilizada, seleção de cores, ponteiros e outros elementos visuais, é definido pelo agente.

Misto: Parte das funções acima são predefinidas em código ou prompt de sistema.

Determinístico: A consulta ao banco de dados, com colunas fixas e predefinidas, é feita de forma programática, assim como o uso de filtros e/ou agregações. A geração do gráfico é feita por script/funções Python.

Observação: Foi experimentado unificar o Chart Calculator e o Chart Designer, mas ao

longo das experimentações, percebeu-se que dividir em duas tarefas (e com o uso adequado de prompts de sistema) facilitou a depuração do código e diminuiu o número de bugs. Além disso, a quebra de responsabilidades está alinhada com a proposta de desenvolvimento.

Busca Web de Notícias

Agêntico: Utiliza agentes que usam ferramentas que acessam a web para buscar por notícias.

Determinístico: Webscraping + análise usando técnicas clássicas de NLP. Apesar de possível, é programaticamente mais complexo e o resultado ainda pode ser tão inesperado quanto uma ferramenta não sofisticada (como DuckDuckGo) ou não se comparar com uma ferramenta sofisticada (Tavily).

Sintetizador de resultados

Agêntico: Utiliza agentes que interpretam textos e números utilizando o potencial de uma LLM.

Determinístico: Dependendo do tipo de dado que se analise, a tarefa pode ser muito complexa para ser feita de forma determinística.

Poppins normal 12pt, para textos longos Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat.

Governança e Observabilidade

Para implementar governança e observabilidade, foram adicionadas 3 funcionalidades. Duas delas se tratam de frameworks amplamente utilizadas, e a última adiciona algumas informações sobre o tema no próprio relatório.

Langsmith

Langsmith é uma plataforma de observabilidade focada em depuração, avaliação e rastreamento de aplicações com LLMs, ajudando times a entender, testar e melhorar prompts, cadeias e agentes com precisão.

Ela é a mais simples de ser implementada, e de fato, não precisa de nenhuma alteração no código. Basta setar a API e verificar os resultados online. No entanto, existe um custo envolvido caso o tamanho da aplicação escale.

Abaixo, temos algumas imagens exemplificando como esta ferramenta ilustra os experimentos executados.

<input type="checkbox"/>	<input checked="" type="checkbox"/>	Name	Input	Output	Error	Start Time	Latency	Tokens	Cost
<input type="checkbox"/>	<input checked="" type="checkbox"/>	LangGraph	Create Full Report	/home/do...		5 minutes ago	🕒 51.05s	20,805	\$0.0587125

TRACE

LangGraph

intent_agent
1.49s 299

gpt-4o
1.39s 299

dispatcher 0.22s

route_based_on_intent 0.07s

chart_calculator
15.87s 7,514

SQL Agent Executor
14.13s 7,514

gpt-4o
0.80s 1,058

sql_db_list_tables 0.00s

gpt-4o
1.53s 1,089

sql_db_schema 0.00s

LangGraph

Run Feedback Metadata

Input

User Prompt

Create Full Report

Branches Completed []

Chart Data

Charts Html

Include Charts false

Include Metrics false

Include News false

Metrics

News Snippets

Raw Data

DT NOTIFIC

EVOLUCAO

UTI

START TIME

01/21/2026, 09:25:27 AM GMT-3

END TIME

01/21/2026, 09:26:18 AM GMT-3

TIME TO FIRST TOKEN

🕒 4.55s

STATUS

🟢 Success

TOTAL TOKENS

20,805 tokens / \$0.0587125

LATENCY

🕒 51.05s

TYPE

Chain

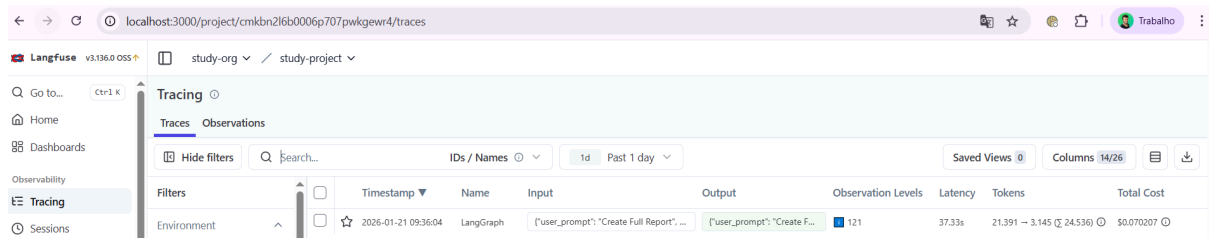
Polly

Langfuse

Similar ao Langsmith, Langfuse também é uma ferramenta de observabilidade e analytics para produtos baseados em LLMs, oferecendo rastreamento, métricas, feedback de usuários e insights para otimizar qualidade e desempenho em produção.

No entanto, é possível utilizar Langfuse de forma local, como pode ser visto abaixo:

```
dougsgrott-wsl@PC-0201: ~/projects/langfuse
langfuse-worker_1 | 2026-01-21T12:34:30.233Z info notification-queue execu
tor started: true
langfuse-worker_1 | 2026-01-21T12:34:30.278Z info Listening: http://2ce137
727b55:3030
langfuse-web_1 | Running init scripts...
redis_1 | 1:M 21 Jan 2026 12:34:30.719 * 1 changes in 3600 seconds. S
aving...
redis_1 | 1:M 21 Jan 2026 12:34:30.721 * Background saving started by
pid 6686
redis_1 | 6686:C 21 Jan 2026 12:34:30.727 * DB saved on disk
redis_1 | 6686:C 21 Jan 2026 12:34:30.727 * Fork CoW for RDB: current
0 MB, peak 0 MB, average 0 MB
redis_1 | 1:M 21 Jan 2026 12:34:30.823 * Background saving terminated
with success
langfuse-worker_1 | 2026-01-21T12:34:30.849Z info [Background Migration] N
o background migrations to run
langfuse-worker_1 | 2026-01-21T12:34:30.855Z info Finished upserting Langf
use dashboards and widgets in 1496ms
langfuse-worker_1 | 2026-01-21T12:34:30.859Z info Finished upserting Langf
use dashboards and widgets in 1498ms
langfuse-worker_1 | 2026-01-21T12:34:30.870Z info Finished upserting defau
lt model prices in 1522ms
langfuse-web_1 | ✓ Ready in 6.4s
langfuse-web_1 | 2026-01-21T12:34:40.602Z info MCP feature registered:
prompts
```



Mesmo sendo open source e rodando localmente, Langfuse também permite a avaliação de variáveis, chamadas a funções, latência, custos, consumo de tokens, etc. Para aplicações onde não é de interesse que os dados sejam enviados para a nuvem, como em muitas aplicações industriais, Langfuse pode ser uma solução de governança e observabilidade com bastante potencial.


Tracing										
Traces		Observations								
Hide filters		Search...		IDs / Names		1d Past 1 day		Saved Views 0	Columns 16/29	
Filters		Start Time ▼	Type	Name	Input	Output	Level	Latency	Total Cost	
Environment Mode: <div>SELECT TEXT</div>	<input type="checkbox"/>	2026-01-21 09:36:42		generate_final_report	{'metrics': {'mortality_rate': 12.14, 'icu_rate': 36.93, 'v...'}	Report successfully saved to: /home/d...	DEFAULT	0.01s	\$0.00	<input type="radio"/>
	<input type="checkbox"/>	2026-01-21 09:36:42		report_maker	{'user_prompt': 'Create Full Report', 'raw_data': {'is_...}}	{'final_report_path': '/home/dougsgr...'}	DEFAULT	0.01s	\$0.00	<input type="radio"/>
	<input type="checkbox"/>	2026-01-21 09:36:38		ChatOpenAI	"\nPerform a comprehensive analysis based on the fol...	"json\n(\n\n`executive_summary`: {\n...	DEFAULT	4.15s	\$0.003548	<input type="radio"/>
	<input checked="" type="checkbox"/> default	2026-01-21 09:36:38		synthesis_agent	{'user_prompt': 'Create Full Report', 'raw_data': {'is_...}}	{'synthesis_state': {'synthesis_result': {...}}	DEFAULT	4.16s	\$0.00	<input type="radio"/>
Type	<input type="checkbox"/>	2026-01-21 09:36:38		barrier_check	{'user_prompt': 'Create Full Report', 'raw_data': {'is_...}}	synthesis_agent	DEFAULT	0.00s	\$0.00	<input type="radio"/>
Name	<input type="checkbox"/>	2026-01-21 09:36:38		mark_charts	{'user_prompt': 'Create Full Report', 'raw_data': {'is_...}}	{'branches_completed': {'charts'}}]	DEFAULT	0.00s	\$0.00	<input type="radio"/>
Name Mode: <div>SELECT TEXT</div>	<input type="checkbox"/>	2026-01-21 09:36:34		ChatOpenAI	"\nGenerate a **standalone HTML snippet** containin...	<div id=\`chart_fdibe88d165d4d568...	DEFAULT	3.61s	\$0.005193	<input type="radio"/>
	<input checked="" type="checkbox"/> ChatOpenAI	2026-01-21 09:36:26		barrier_check	{'user_prompt': 'Create Full Report', 'raw_data': {'is_...}}	__end__	DEFAULT	0.00s	\$0.00	<input type="radio"/>
Trace Name	<input type="checkbox"/>	2026-01-21 09:36:26		mark_news	{'user_prompt': 'Create Full Report', 'raw_data': {'is_...}}	{'branches_completed': {'news'}}]	DEFAULT	0.00s	\$0.00	<input type="radio"/>
Level	<input type="checkbox"/>	2026-01-21 09:36:26		barrier_check	{'user_prompt': 'Create Full Report', 'raw_data': {'is_...}}	__end__	DEFAULT		\$0.00	<input type="radio"/>
Model	<input type="checkbox"/>	2026-01-21 09:36:26		mark_metrics	{'user_prompt': 'Create Full Report', 'raw_data': {'is_...}}	{'branches_completed': {'metrics'}}]	DEFAULT	0.00s	\$0.00	<input type="radio"/>
Model ID	<input type="checkbox"/>	2026-01-21 09:36:26		ChatOpenAI	"\nGenerate a **standalone HTML snippet** containin...	<div id=\`chart_3ceca3ac938a4b478a...	DEFAULT	8.16s	\$0.008588	<input type="radio"/>

Por fim, como o resultado final é um relatório, foi decidido adicionar elementos relacionados a governança no próprio relatório. Tais elementos incluem a data em que o relatório foi gerado, o prompt utilizado, se o guardrail foi ativado e quais branches foram executadas (todas, por padrão, devido a forma que a estrutura foi construída).

⚠ Notice: Prompt 'Qual premio O Agente Secreto Ganhou?' was off-topic. Standard report generated.

Generation Timestamp: 2026-01-21 (UTC)

Execution Path: ✓ Metrics Analyst
✓ Chart Calculator
✓ News Researcher

 **AUDITING & GOVERNANCE**

Generation Timestamp: 2026-01-12 (UTC)

Execution Path: ✔ Metrics Analyst
✔ Chart Calculator
✔ News Researcher

Relatório

Finalmente, como o objetivo do projeto é a construção do relatório, esta seção visa ilustrar como o relatório foi construído.

Título e autoria:

Contém o título, data de criação e autoria do relatório

SARS Outbreak Status Report - 2026-01-21

Prepared by: Indicium HealthCare Agent

Métricas:

Inclui as principais métricas requeridas no enunciado do projeto.

Key Statistical Metrics

Rate of Increase (30 Days):	Mortality Rate:	ICU Occupancy:	Vaccination Rate:
0.0%	12.14%	36.93%	32.09%

Síntese:

A síntese leva em consideração tanto as métricas, como os dados de tendência do gráfico e as notícias da web.

Scenario Context & Commentary

Executive Summary

The current epidemiological situation appears stable with no significant increase in daily cases over the past 30 days. However, the high mortality and ICU rates suggest underlying challenges in healthcare management. The low vaccination rate may be contributing to these adverse outcomes.

Risk Assessment

{'threat_level': 'Moderate', 'recommendations': ['Increase vaccination efforts to improve coverage and reduce severe outcomes.', 'Enhance healthcare capacity and resources to manage high ICU and mortality rates effectively.']}

Contextual Deep Dive

The mortality rate is notably high at 12.14%, which could indicate severe cases or inadequate healthcare response, although no news reports are available to confirm this. The ICU rate of 36.93% is also concerning, suggesting a significant portion of cases require intensive care, potentially straining healthcare resources. The vaccination rate is relatively low at 32.09%, which may be insufficient to prevent severe outcomes and could explain the high mortality and ICU rates. The stable daily cases trend, with no increase over the last 30 days, suggests that transmission is currently under control, but the lack of news makes it difficult to ascertain the reasons behind these trends.

Notícias:

A seção de notícias contém um resumo e um link das notícias encontradas. Pelas experimentações realizadas, é comum que o DuckDuckGo não encontre notícias relevantes ou estranhos artigos escritos em outras línguas como arábico. Usando o

Tavily, no entanto, o número de notícias relevantes é bem maior.

Supporting News Snippets:

- [Raw Search Output](#)

No good DuckDuckGo Search Result was found

Gráficos:

Esta seção contém os gráficos gerados.

Case Trend Visualizations



Governança e observabilidade:

Por fim, como mostrado na seção anterior, também foi adicionado elementos de governança e observabilidade no relatório.

AUDITING & GOVERNANCE

Generation Timestamp:

2026-01-21 (UTC)

User Prompt:

"Create Full Report"

Guardrail Status:

PASSED

Execution Path:

✓ Metrics Analyst

✓ Chart Calculator

✓ News Researcher

Trace ID: 18c42ac5 | Engine: Indicium SARS-Lens v1.0

Executando o projeto

Requisitos do projeto

Para a execução do projeto, se faz necessário cumprir alguns requisitos obrigatórios:

- Python (recomendado: 3.12)
- uv como Python Package Manager
- Uma chave para o uso de OpenAI

Além disso, o projeto também tem alguns requisitos **opcionais**, relacionados a melhorias de performance e/ou observabilidade e governança:

- Chave para o uso de Tavily (pago, mas com free tier)
- Chave para o uso de Langsmith (pago, mas com free tier)
- Chave para o uso de Langfuse (pago, mas com free tier e opção de rodar localmente)

Caso Tavily não possa ser utilizado, o buscador DuckDuckGo será utilizado por padrão.

Execução

Dentro da pasta scripts, existe um script em Python chamado run_report_maker.py, que permite ao usuário selecionar uma de três opções:

- 1) Linear Workflow (Waterfall – Deterministic)
 - Se refere ao workload linear apresentado anteriormente.
- 2) Parallel Workflow (Fan-Out/Fan-In – Deterministic)
 - Se refere ao workload paralelo sem detecção de intenção do prompt.
- **3) Conditional Workflow (Intent-Based – User Input)**
 - **RECOMENDADO.** Se refere ao workload final e mais testado, com detecção de intenção e paralelismo.

Quando a **opção 3** é selecionada, é possível fornecer um prompt. Para criar um relatório completo, basta digitar "Create full report" ou algo do tipo. Este procedimento é ilustrado abaixo:

```
(.venv) dougsgrott-wsl@PC-0201:~/projects/indiciu_ai_eng$ python ./scripts/run_report_maker.py

=====
EXECUTING END-TO-END SARS REPORT PIPELINE
=====

Select Execution Mode:
1. Linear Workflow (Waterfall - Deterministic)
2. Parallel Workflow (Fan-Out/Fan-In - Deterministic)
3. Conditional Workflow (Intent-Based - User Input)

Enter choice [1-3]: 3
>> Selected: Conditional Workflow
>> Enter your request (e.g., 'Only show me news about vaccines'): Create Full Report
```

Resultado

O resultado da execução deve ser um relatório, conforme ilustrado abaixo:

SARS Outbreak Status Report - 2026-01-21

Prepared by: Indiciu HealthCare Agent

Key Statistical Metrics

Rate of Increase (30 Days):	Mortality Rate:	ICU Occupancy:	Vaccination Rate:
0.0%	12.14%	36.93%	32.09%

Scenario Context & Commentary

Executive Summary

The current epidemiological situation appears stable with no significant increase in daily cases over the past 30 days. However, the high mortality and ICU rates suggest underlying challenges in managing severe cases. The low vaccination rate may be contributing to these severe outcomes.

Risk Assessment

{'threat_level': 'High', 'recommendations': ['Increase vaccination efforts to improve coverage and reduce severe outcomes.', 'Conduct a thorough investigation into healthcare system capacity and resource allocation to address high ICU and mortality rates.']}

Contextual Deep Dive

The mortality rate of 12.14% and ICU rate of 36.93% are notably high, indicating a significant burden on healthcare resources, despite the stable daily case numbers. The lack of increase in daily cases suggests effective containment or reporting issues, but the low vaccination rate of 32.09% could be a factor in the high severity of cases. Without recent news, it's challenging to pinpoint specific causes, but these metrics suggest potential healthcare system strain.

Supporting News Snippets:

Raw Search Output

No good DuckDuckGo Search Result was found

Case Trend Visualizations

Casos Diários (Últimos 30 Dias)

Day	Cases
1	1.0