UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Chatbot baseado em *Large Language Models* (LLMs) e *Fast Healthcare Interoperability Resources* (FHIR) para monitoramento de indivíduos com risco para o câncer de boca

Renata Dutra Braga

Monografia - MBA em Inteligência Artificial e Big Data

ICMC USP SÃO CARLOS

Renata Dutra Braga

Chatbot baseado em *Large Language Models* (LLMs) e *Fast Healthcare Interoperability Resources* (FHIR) para monitoramento de indivíduos com risco para o câncer de boca

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientadora: Profa. Dra. Carolina Toledo Ferraz

Versão original

São Carlos

2025

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi, ICMC/USP, com os dados fornecidos pelo(a) autor(a)

S856m	Braga, Renata Dutra *Chatbot* baseado em *Large Language Models* (LLMs) e *Fast Healthcare Interoperability Resources* (FHIR) para monitoramento de indivíduos com risco para o câncer de boca / Renata Dutra Braga ; orientadora Carolina Toledo Ferraz. – São Carlos, 2025. 82 p. : il. (algumas color.) ; 30 cm. Monografia (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2025. 1. Chatbots em saúde. 2. Grandes Modelos de Linguagem. 3. FHIR. 4. Câncer de boca. 5. Interoperabilidade semântica em saúde. 6. Saúde digital. I. Ferraz, Carolina Toledo, orient. II. Título.

Renata Dutra Braga

Chatbot Based on Large Language Models (LLMs) and Fast Healthcare Interoperability Resources (FHIR) for Monitoring Individuals at Risk for Oral Cancer

Monograph presented to the Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, as part of the requirements for obtaining the title of Specialist in Artificial Intelligence and Big Data.

Concentration area: Artificial Intelligence

Advisor: Profa. Dra. Carolina Toledo Ferraz

Original version

São Carlos

2025

Dedico este trabalho ao riso leve dos meus filhos, Henrique e Cecília, que enchem de cor os meus dias.

Ao abraço constante do meu marido, Leandro, porto seguro em todas as tempestades.

Ao cuidado e ensinamentos dos meus pais, Vilmar e Valdivina, que moldaram a minha essência.

Ao apoio de toda a minha família, irmãos, sogra, sogro, cunhados, sobrinhos e amigos.

E a Deus, pela dádiva da vida e pela coragem de seguir sempre em frente.

AGRADECIMENTOS

Agradeço, primeiramente, a Deus, pela saúde, sabedoria e força para enfrentar os desafios desta caminhada.

À minha orientadora, Profa. Dra. Carolina Toledo Ferraz, pela confiança, dedicação, paciência e contribuições para a realização deste trabalho.

À minha família, especialmente a meu marido Leandro, companheiro constante, e a meus filhos, Henrique e Cecília, que são fonte de motivação e inspiração.

A meus pais, Vilmar e Valdivina, pelo apoio incondicional, pelos ensinamentos transmitidos e pelo exemplo de dedicação e perseverança. Além deles, não poderia deixar de agradecer aos meus irmãos, sobrinhos(as), cunhados(as), sogra (Heleni), ao meu sogro (Eloi) e amigos.

Aos colegas, tutores e professores do Curso, pela troca de experiências, incentivo e pela construção coletiva do conhecimento.

"Você não consegue ligar os pontos olhando para frente; só é possível conectá-los olhando para trás. Por isso, você precisa confiar que os pontos, de alguma forma, se ligarão no futuro."

Steve Jobs

(Discurso na cerimônia de formatura na Universidade de Stanford, em 2005.)

Esta citação ressalta a importância de confiar no processo da vida, de seguir a intuição e de acreditar que as experiências passadas se conectam para dar sentido e propósito ao futuro.

RESUMO

BRAGA, R.D. ***Chatbot* baseado em *Large Language Models* (LLMs) e *Fast Healthcare Interoperability Resources* (FHIR) para monitoramento de indivíduos com risco para o câncer de boca**. 2025. 82 p. Monografia (MBA em Inteligência Artificial e Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2025.

O avanço das tecnologias de linguagem natural e dos padrões de interoperabilidade em saúde tem aberto novas possibilidades para a coleta estruturada de informações clínicas. Contudo, a integração entre *chatbots* inteligentes e o padrão HL7® FHIR®, ainda apresenta desafios técnicos e semânticos, especialmente em cenários de atenção primária e vigilância epidemiológica. Nesse contexto, surgiu a necessidade de desenvolver uma solução capaz de transformar interações em linguagem natural em registros clínicos interoperáveis, de modo a apoiar o rastreamento e o monitoramento da população de risco para o câncer de boca. O objetivo geral deste estudo foi desenvolver agentes inteligentes baseados em LLMs, integrados ao padrão HL7® FHIR®, capazes de orquestrar interações personalizadas e interoperáveis para coleta, estruturação e análise de informações clínicas, apoiando o rastreamento e monitoramento desta população de risco. A metodologia adotada foi de natureza aplicada e experimental, organizada em quatro etapas. A primeira etapa consistiu na modelagem da informação clínica em FHIR®, definindo elementos de dados (variáveis) e regras necessárias. A segunda etapa envolveu o desenvolvimento dos agentes inteligentes (Orquestração, Perguntas e Interação), baseados em LLMs integrados ao *framework LangChain* e conectados ao banco de dados MongoDB. A terceira etapa tratou da implementação da interface conversacional, desenvolvida em *Streamlit* e integrada ao *WhatsApp* via *EvolutionAPI*, com *webhooks* para comunicação em tempo real. Por fim, a quarta etapa compreendeu a avaliação do *pipeline* conversacional, verificando a coerência entre os módulos, a integridade dos dados e a persistência dos recursos no servidor HAPI FHIR. Os resultados demonstraram que a solução foi capaz de converter mensagens em linguagem natural em recursos FHIR computáveis, interoperáveis e semanticamente consistentes. A solução manteve memória de sessão, normalizou dados de forma automática e garantiu a persistência bem-sucedida no servidor HAPI FHIR. As validações comprovaram a completude e a integração do fluxo conversacional, evidenciando a confiabilidade dos agentes na coleta estruturada e validação dos dados clínicos, tanto na interface *web* quanto no canal *WhatsApp*. A arquitetura desenvolvida comprovou ser tecnicamente viável para coleta estruturada de informações clínicas via diálogo natural. A união entre orquestração determinística, inferência baseada em LLMs e validação de regras FHIR mostrou-se eficaz para garantir consistência e interoperabilidade dos dados. Embora limitada a informações cadastrais do paciente, a solução representa uma prova de conceito sólida, com potencial

de ampliação para outros domínios clínicos e integração à Rede Nacional de Dados em Saúde (RNDS).

Palavras-chave: Agentes inteligentes. Grandes Modelos de Linguagem. Coleta conversacional. Saúde digital. HL7 FHIR. Interoperabilidade semântica.

ABSTRACT

BRAGA, R.D. **Chatbot Based on Large Language Models (LLMs) and Fast Healthcare Interoperability Resources (FHIR) for Monitoring Individuals at Risk for Oral Cancer**. 2025. 82 p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2025.

The advancement of natural language technologies and health interoperability standards has opened new possibilities for the structured collection of clinical information. However, the integration between intelligent chatbots and the HL7® FHIR® standard still poses technical and semantic challenges, particularly in primary care and epidemiological surveillance contexts. In this scenario, there emerged the need to develop a solution capable of transforming natural language interactions into interoperable clinical records, thereby supporting the screening and monitoring of populations at risk for oral cancer. The main goal of this study was to develop intelligent agents based on large language models (LLMs), integrated with the HL7® FHIR® standard, capable of orchestrating personalized and interoperable interactions for the collection, structuring, and analysis of clinical information, supporting the tracking and monitoring of this at-risk population. The adopted methodology was applied and experimental in nature, organized into four stages. The first stage involved modeling clinical information according to the HL7® FHIR® standard, defining data elements (variables) and interoperability rules. The second stage comprised the development of intelligent agents (Orchestration, Question, and Interaction Agents), built upon LLMs integrated with the LangChain framework and connected to a MongoDB database. The third stage addressed the implementation of the conversational interface, developed in Streamlit and integrated with WhatsApp via EvolutionAPI, using webhooks for real-time communication. Finally, the fourth stage encompassed the evaluation of the conversational pipeline, verifying module coherence, data integrity, and the persistence of resources in the HAPI FHIR server. The results demonstrated that the proposed solution successfully converted natural language messages into computable, interoperable, and semantically consistent FHIR resources. The system maintained session memory, automatically normalized data, and ensured successful persistence in the HAPI FHIR server. Validation tests confirmed the completeness and integration of the conversational flow, evidencing the reliability of the agents in structured data collection and clinical data validation, both in the web interface and the WhatsApp channel. The developed architecture proved to be technically feasible and methodologically coherent for structured collection of clinical information through natural dialogue. The combination of deterministic orchestration, LLM-based inference, and FHIR rule validation proved effective in ensuring data consistency and interoperability. Although limited to basic patient demographic information, the solution represents a solid proof of concept with potential for expansion

to other clinical domains and integration into the Brazilian National Health Data Network (Rede Nacional de Dados em Saúde – RNDS).

Keywords: Intelligent agents. Large Language Models; Conversational data collection. Digital health. HL7® FHIR®. Semantic interoperability.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

APS	Atenção Primária à Saúde
CB	Câncer de Boca
CEP	Código de Endereçamento Postal
CNS	Consulta de Profissionais de Saúde
CPF	Cadastro de Pessoas Físicas
FHIR	*Fast Healthcare Interoperability Resources*
FSH	FHIR *Shorthand*
IA	Inteligência Artificial
LGPD	Lei Geral de Proteção de Dados
LLMs	*Large Language Models*
MI	Modelo de informação
PLN	Processamento de Linguagem Natural
RAG	*Retrieval-Augmented Generation*
SUS	Sistema Único de Saúde

SUMÁRIO

1 INTRODUÇÃO

O presente estudo apresenta o desenvolvimento e a validação de uma arquitetura inteligente voltada à coleta e estruturação de informações clínicas em linguagem natural, integrando Grandes Modelos de Linguagem (*Large Language Models* – LLMs) ao padrão de interoperabilidade HL7® FHIR®. A proposta insere-se no contexto da saúde digital e visa apoiar o rastreamento e o monitoramento de populações de risco para o câncer de boca, fortalecendo a integração entre tecnologias de inteligência artificial e sistemas de informação em saúde.

1.1 Contextualização e Problema

O câncer de boca (CB) apresenta alta incidência em populações de risco, como homens com idade acima de 40 anos, tabagistas, etilistas, com exposição solar frequente, entre outros, e sua detecção precoce é fundamental para aumentar as chances de tratamento e cura (Speight *et al.*, 2017). Contudo, a falta de ferramentas acessíveis para rastreamento e monitoramento dessas populações dificulta o reconhecimento de sinais precoces, comprometendo a efetividade das ações para prevenção do CB.

Apesar do potencial das tecnologias de inteligência artificial, ainda são escassas as soluções que utilizam *Large Language Models* (LLMs) para oferecer suporte ao rastreamento e monitoramento dessas populações (Claman; Sezgin, 2024; Giannakopoulos *et al.*, 2023), combinando a capacidade de linguagem natural dos LLMs com a coleta estruturada de dados clínicos de forma interoperável. Além disso, a modelagem estruturada de informações clínicas, como sintomas, fatores de risco e histórico clínico, em conformidade com padrões interoperáveis, como o *Fast Healthcare Interoperability Resources* (FHIR), é frequentemente negligenciada em sistemas de informação em saúde, limitando a reutilização e a integração dos dados entre diferentes sistemas de informação em saúde (Schmiedmayer *et al.*, 2024).

Paralelamente, há uma lacuna na estruturação das informações clínicas coletadas em sistemas de saúde. Dados sobre o CB são frequentemente registrados de maneira fragmentada, dificultando a sua reutilização para vigilância epidemiológica, integração com outros sistemas e continuidade do cuidado. O uso de padrões como o FHIR é fundamental para garantir a padronização e interoperabilidade dessas informações, mas ainda é pouco explorado em soluções baseadas em IA (Schmiedmayer *et al.*, 2024).

Neste contexto, fica evidenciada a necessidade de desenvolver um *chatbot* inteligente que, ao integrar recursos de LLMs, permita coletar e organizar informações clínicas estruturadas no padrão FHIR. Com isso, buscou-se não apenas rastrear indivíduos, alertando para sinais de risco que demandem avaliação clínica pelo dentista, mas também criar uma base

de dados alinhada à Estratégia de Saúde Digital para o Brasil (2020-2028), promovendo a interoperabilidade e rastreamento das populações de risco para CB (Brasil, 2020).

1.2 Justificativa e Motivação

O CB é um dos principais desafios oncológicos no Brasil, com 15.100 novos casos anuais estimados entre 2023 e 2025, sendo o quarto mais frequente entre homens na Região Sudeste (13,16 por 100 mil) (INCA, 2023; Santos *et al.*, 2023). Associado a fatores como tabagismo, consumo excessivo de álcool e exposição solar prolongada, esse câncer apresenta elevada incidência e prognóstico desfavorável devido à detecção tardia, barreiras no acesso ao tratamento e lacunas na educação em saúde, especialmente na Atenção Primária à Saúde (APS), principal porta de entrada do Sistema Único de Saúde (SUS).

A aplicação de LLMs oferece uma solução inovadora para enfrentar esses desafios por meio de um *chatbot* inteligente capaz de identificar indivíduos que pertencem ao grupo de risco (rastreamento), interagindo de forma natural. Essa tecnologia promove a educação, monitoramento e suporte contínuo de forma acessível e adaptada às necessidades dos usuários, beneficiando tanto a população quanto os profissionais de saúde (Albert; Tizzard, 2024; Aydin *et al.*, 2024). Além disso, contribui com a Prioridade 7 da Estratégia de Saúde Digital para o Brasil, que visa criar um ecossistema de inovação voltado ao uso secundário de dados e desenvolvimento de novas tecnologias (Brasil, 2020).

A modelagem estruturada de informações clínicas no padrão FHIR é uma etapa essencial para garantir a organização e reutilização dos dados coletados durante o rastreamento. Essa solução possibilita identificar se um indivíduo pertence ao grupo de risco para CB, além de preparar as informações para integração futura com outros sistemas, como o projeto SobreVidas (Pedrosa *et al.*, 2024). Assim, o desenvolvimento de um *chatbot* baseado em LLMs, que integre modelagem de dados estruturados, representa um avanço importante para a saúde digital no Brasil. Essa solução potencializa a detecção precoce, melhora a organização dos dados clínicos e fortalece a APS como ponto central de promoção, prevenção e cuidado em saúde.

1.3 Questões de Pesquisa e Objetivos

Diante dos desafios identificados, formulou-se a seguinte questão de pesquisa: Como a integração de agentes inteligentes baseados em LLMs e modelos de informação estruturados no padrão FHIR podem contribuir para melhorar a efetividade do rastreamento e monitoramento da população de risco para o CB?

Com base nessa questão, definiu-se como objetivo geral: Desenvolver agentes inteligentes baseados em LLMs, integrados ao padrão HL7® FHIR®, capazes de orquestrar interações personalizadas e interoperáveis para coleta, estruturação e análise de informações

clínicas, apoiando o rastreamento e monitoramento da população de risco para o CB e a integração com sistemas de saúde digital.

Os seguintes objetivos específicos foram delineados:

- Modelar informações clínicas e demográficas de forma estruturada e interoperável, utilizando o padrão FHIR para representar diferentes tipos de registros.
- Implementar agentes inteligentes baseados em LLMs e arquiteturas multiagente para conduzir interações naturalizadas, validando e normalizando dados clínicos de forma automatizada.
- Desenvolver mecanismos de orquestração e persistência capazes de transformar as interações em recursos FHIR computáveis, garantindo rastreabilidade, auditabilidade e interoperabilidade dos dados.
- Desenvolver uma interface conversacional multiplataforma para a comunicação entre os agentes e os usuários.

1.4 Organização do Trabalho

- **Capítulo 1 – Introdução:** apresenta a contextualização do problema, a justificativa e a relevância do estudo, além da formulação da questão de pesquisa, dos objetivos gerais e específicos e da organização deste estudo.
- **Capítulo 2 – Fundamentação Teórica:** discute os principais conceitos e estudos relacionados à pesquisa, abordando os temas: linguagem natural, agentes inteligentes, interoperabilidade em saúde, o padrão HL7® FHIR® e suas aplicações na saúde digital. Este capítulo estabelece o embasamento teórico que fundamenta o desenvolvimento do modelo proposto.
- **Capítulo 3 – Metodologia:** descreve o percurso metodológico adotado, detalhando o tipo de pesquisa, as etapas de desenvolvimento e os recursos tecnológicos utilizados. Explicita a modelagem da informação clínica, a arquitetura multiagente, os componentes do *pipeline* conversacional e os mecanismos de integração e validação.
- **Capítulo 4 – Avaliação Experimental:** apresenta os resultados obtidos com o modelo desenvolvido, incluindo os testes realizados, a análise funcional e técnica do *pipeline* conversacional e a validação da interoperabilidade dos recursos FHIR. Este capítulo também demonstra a aplicação prática dos agentes inteligentes e da interface conversacional.

- **Capítulo 5 – Conclusões:** reúne as conclusões da pesquisa, discutindo as contribuições alcançadas, as limitações identificadas e as perspectivas de evolução da solução, destacando oportunidades de expansão para novos domínios clínicos.

2 FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

A fundamentação teórica deste estudo foi construída a partir de estudos recentes que abordam a interface entre Inteligência Artificial, Saúde Digital e Interoperabilidade em Saúde. Para reforçar o alinhamento cronológico e a atualidade das fontes utilizadas, apresenta-se, a seguir, uma linha do tempo com as principais referências bibliográficas que embasaram o desenvolvimento da pesquisa.

A Tabela 1 organiza as referências por ano de publicação, destacando o tema principal abordado em cada estudo, evidenciando a evolução dos conceitos-chave do projeto (desenvolvimento dos LLMs; avanço da interoperabilidade baseada no padrão FHIR; e o crescimento das aplicações de *chatbots* inteligentes na saúde).

A utilização de referências atualizadas promove a aderência deste estudo às tendências mais recentes em inovação tecnológica na área da saúde, bem como a sua consonância com as diretrizes nacionais e internacionais para transformação digital em saúde (Brasil, 2020; WHO, 2021).

Tabela 1 – Linha do tempo das referências bibliográficas.

Ano	**Referência**	**Tema Principal**
2016	(Mandel *et al.*, 2016)	Interoperabilidade *SMART on FHIR*
2017	(Vaswani *et al.*, 2017)	Transformer: Arquitetura para LLMs
2018	(Hussain; Langer; Kohli, 2018)	Validação e implementação do padrão FHIR
2018	(Laranjo *et al.*, 2018)	*Chatbots* na Saúde: Revisão sistemática
2019	(Topol, 2019)	Inteligência Artificial e Humanização da Saúde
2020	(Brown *et al.*, 2020)	GPT-3: Modelos de linguagem como few-shot learners
2020	(Meskó; Görög, 2020)	Inteligência Artificial em Saúde: Guia para médicos
2021	(Sato *et al.*, 2021)	Viabilidade de um *chatbot* baseado em inteligência aumentada para realizar a triagem preliminar de câncer
2022	(Shah; Khan, 2020)	Integração de FHIR com *big data* e IA
2023	(Albert; Tizzard, 2024)	Aplicações de LLMs na saúde
2023	(Aydin *et al.*, 2024)	Rastreamento digital com LLMs

Ano	Referência	Tema Principal
2023	(Bubeck *et al.*, 2023)	Experimentos com GPT-4 e emergências de inteligência geral
2023	(Thirunavukarasu *et al.*, 2023)	Uso de LLMs em aplicações clínicas
2023	(Giannakopoulos *et al.*, 2023)	LLMs na triagem de atenção primária
2023	(Ji *et al.*, 2023)	Estudo sobre alucinações em LLMs
2023	(Obermeyer *et al.*, 2019)	Análise de viés algorítmico em saúde
2023	(Santos *et al.*, 2023)	Estimativas de câncer de boca no Brasil
2023	(Brasil, 2020)	Estratégia de Saúde Digital para o Brasil 2020-2028
2023	(HL7 International, 2023)	Padrão FHIR *Release* 4 - Especificação Técnica
2024	(Schmiedmayer *et al.*, 2024)	Aplicação de FHIR em soluções de IA
2025	(Altom *et al.*, 2025)	*Chatbots* na gestão de doenças crônicas
2025	(Singhal *et al.*, 2025)	Med-PaLM: Respostas médicas com LLMs
2025	(Oshin; Campos, 2025)	Construção de agentes inteligentes com *LangChain*

2.1 LLMs na saúde: fundamentos teóricos e aplicações clínicas

Os LLMs representam um paradigma transformador no campo da inteligência artificial, redefinindo as possibilidades de interação homem-máquina através de avanços em Processamento de Linguagem Natural (PLN). Esses sistemas, arquitetonicamente fundamentados em redes neurais profundas do tipo *transformer*, caracterizam-se por sua capacidade de processamento contextualizado de linguagem, possibilitado pela análise de padrões em corpus textuais massivos (Vaswani *et al.*, 2017). Modelos de última geração, como o GPT-4 da OpenAI, PaLM 2 do Google e LLaMA 2 da Meta, demonstram desempenho comparável a benchmarks humanos em diversas tarefas linguísticas, graças a arquiteturas que podem englobar até 1 trilhão de parâmetros e treinamento em exabytes de dados multimodal (Bubeck *et al.*, 2023).

No domínio da saúde, os LLMs emergem como tecnologias disruptivas com potencial para reconfigurar os processos assistenciais, educacionais e de gestão. A sua aplicação transcende a geração textual, abrangendo: (1) análise semântica de prontuários eletrônicos, (2) síntese de evidências científicas, (3) apoio à decisão clínica, e (4) interação naturalizada com usuários de sistemas de saúde. Estudos recentes demonstram que modelos *fine-tuned*

para domínios médicos, como o Med-PaLM (Singhal *et al.*, 2025), alcançam desempenho similar a médicos humanos em exames de certificação médica, sugerindo potencial para aplicações clinicamente relevantes.

A transição dos *chatbots* baseados em regras (*Rule Based*) para sistemas fundamentados em LLMs representa uma mudança de paradigma na computação conversacional. Enquanto os primeiros operam em espaços de ação limitados, com diálogos pré-definidos e reconhecimento restrito de intenções, os LLMs habilitam interações contextualmente adaptáveis, capazes de: (i) lidar com variações linguísticas naturais, (ii) inferir intenções implícitas, e (iii) personalizar respostas conforme o perfil sociocultural do usuário (Ji *et al.*, 2023). Essa capacidade é importante em contextos de saúde pública, onde a diversidade linguística e educacional da população frequentemente limita a efetividade de soluções tecnológicas tradicionais.

Na APS, os LLMs demonstram potencial para atuar como multiplicadores (*force multipliers*), ampliando a capacidade assistencial dos profissionais de saúde por meio de: (a) triagem automatizada baseada em protocolos clínicos validados (ex.: *Manchester*, *Canadian Triage*), (b) coleta estruturada de dados anamnésicos, e (c) educação em saúde. Contudo, a adoção clínica de LLMs enfrenta desafios que demandam abordagens multidimensionais. A questão da confiabilidade clínica (*clinical trustworthiness*) emerge como ponto central, envolvendo: (1) validação de saídas frente às diretrizes reconhecidas, (2) mitigação de alucinações (*hallucinations*) - fenômeno onde o modelo gera informações factualmente incorretas, e (3) transparência no processo decisório (explicabilidade) (Yang *et al.*, 2023).

Estudos de validação sugerem que mesmo os melhores modelos atuais apresentam taxas de erro clinicamente significativas (5-15% em tarefas diagnósticas), reforçando a necessidade de supervisão humana e sistemas de *fallback* (referem-se a mecanismos alternativos que são acionados quando o sistema principal falha ou não é capaz de fornecer uma resposta adequada, garantindo assim a continuidade do serviço e a segurança da experiência do usuário (Singhal *et al.*, 2025). Em contextos sensíveis, como a saúde, esses mecanismos podem incluir desde redirecionamento para atendimento humano até a consulta a bases estruturadas de conhecimento.

A dimensão ético-legal apresenta igual complexidade, particularmente no que tange à: (i) proteção de dados sensíveis (*compliance* com a LGPD), (ii) responsabilização por erros (*accountability*), e (iii) viés algorítmico. Pesquisas demonstram que LLMs podem perpetuar ou amplificar vieses presentes nos dados de treinamento, como sub-representação de populações específicas ou estereótipos culturais (Obermeyer *et al.*, 2019). Estratégias como *human-in-the-loop* e *Retrieval-Augmented Generation* (RAG) têm sido propostas para mitigar esses riscos, embora soluções definitivas permaneçam como área ativa de pesquisa.

No contexto desta pesquisa, a adoção de LLMs é orientada por um *framework*

de desenvolvimento modular, que integra (1) a modelagem estruturada de informações clínicas no padrão HL7® FHIR®, (2) o desenvolvimento de agentes inteligentes capazes de conduzir interações personalizadas e validar dados clínicos de forma automatizada, (3) a integração com repositórios interoperáveis, como MongoDB e HAPI FHIR *Server*, e (4) a implementação de interfaces conversacionais acessíveis para o público-alvo. Assim, espera-se equilibrar a flexibilidade conversacional dos LLMs com a precisão semântica e clínica necessária em contextos de rastreamento oncológico, especialmente no CB, em que fatores como baixa literacia em saúde e acesso restrito a especialistas comprometem o diagnóstico precoce e a continuidade do cuidado..

A integração proposta com a Estratégia de Saúde Digital para o Brasil (2020–2028) (Brasil, 2020) ocorre em três níveis: (i) técnico (interoperabilidade via FHIR), (ii) organizacional (fluxos assistenciais complementares), e (iii) ético (equidade no acesso). Essa triangulação posiciona este estudo na fronteira das inovações em saúde digital, oferecendo um modelo potencialmente replicável para outras condições crônicas no SUS.

2.2 Padrão FHIR e interoperabilidade em saúde: fundamentos técnicos e aplicações em Saúde Digital

A interoperabilidade em saúde configura-se como um dos principais desafios para a consolidação de sistemas de informação integrados, especialmente em cenários de diversidade tecnológica e fragmentação de sistemas, como é o caso do SUS brasileiro. A ausência de padronização semântica e sintática entre sistemas eletrônicos de saúde dificulta a continuidade do cuidado, a vigilância epidemiológica e a eficácia de políticas públicas baseadas em evidências. Nesse contexto, a adoção de padrões de interoperabilidade torna-se imperativa para superar essas barreiras e viabilizar o fluxo ágil e seguro para trocas de informações clínicas (Mandel *et al.*, 2016).

Dentre os padrões existentes, o FHIR, desenvolvido pela HL7 *International*, destaca-se como o *framework* mais avançado e adotado globalmente para representação e troca de dados em saúde. Baseado em princípios de arquitetura orientada a recursos (RESTful), o FHIR organiza informações em componentes modulares denominados de recursos (*resources*) (exemplos, *Patient*, *Observation*, *Condition*, *MedicationRequest*), que podem ser serializados em diferentes formatos, como JSON e XML. Esse padrão permite não apenas a interoperabilidade técnica (sintática), mas também a interoperabilidade semântica, garantindo que os dados trocados mantenham seu significado clínico em diferentes contextos de uso (HL7 International, 2023).

A relevância do FHIR para este estudo reside em sua capacidade de estruturar dados clínicos, administrativos e epidemiológicos de forma padronizada, facilitando a sua integração com sistemas heterogêneos, tais como: sistemas nacionais de saúde (exemplos: e-SUS e SISAB); plataformas hospitalares e de prontuário eletrônico do paciente; bancos

de dados de pesquisa e análises populacionais; e, ferramentas de inteligência artificial e aprendizado de máquina (Shah; Khan, 2020).

O HAPI FHIR *Server* e as bibliotecas de validação de recursos asseguram a conformidade com as especificações do padrão (Hussain; Langer; Kohli, 2018), promovendo a consistência e a qualidade dos dados. Além disso, o FHIR viabiliza o uso secundário de dados, ampliando as suas aplicações para: (a) análises epidemiológicas e preditivas, permitindo a identificação de tendências e fatores de risco em larga escala; (b) avaliação de políticas públicas, com base em dados estruturados e comparáveis; (c) desenvolvimento de modelos de IA em saúde, onde a padronização dos dados é crítica para treinamento e validação de algoritmos.

Essas capacidades estão alinhadas com a Estratégia de Saúde Digital para o Brasil (2020–2028) (Brasil, 2020), em especial à Prioridade Estratégica 7, que enfatiza a utilização de dados para inovação e fortalecimento da atenção primária. Contudo, apesar de sua crescente adoção em sistemas tradicionais, o FHIR ainda é subutilizado em projetos envolvendo LLMs. A maioria dos assistentes virtuais em saúde opera de forma isolada, sem aderência a padrões de interoperabilidade, o que limita a sua integração com outras plataformas e compromete a reutilização dos dados gerados.

Neste estudo, a incorporação do FHIR desde a fase de concepção do agente conversacional representa um avanço metodológico, unindo a flexibilidade linguística dos LLMs com a estruturação semântica e interoperável dos dados clínicos, garantindo (a) rastreabilidade dos dados, por meio de metadados padronizados que documentam origem, contexto e processamento; (b) auditabilidade e conformidade regulatória, essencial para conformidade com regulamentações como a LGPD; e, (c) sustentabilidade técnica e evolutiva, possibilitando a expansão da solução em ecossistemas complexos de saúde digital e sua integração com outros sistemas e serviços do SUS.

Assim, a sinergia entre LLMs e FHIR transcende o aspecto técnico, configurando-se como uma oportunidade estratégica para impulsionar a saúde digital no Brasil. Ao alinhar tecnologias emergentes com padrões internacionais, este estudo contribui para a construção de soluções escaláveis, interoperáveis e aderentes às diretrizes nacionais e globais de e-Saúde.

2.3 Aplicações de *chatbots* inteligentes no rastreamento de doenças

Os sistemas conversacionais inteligentes, particularmente aqueles fundamentados em modelos avançados de PLN, têm emergido como instrumentos estratégicos no âmbito da saúde digital, oferecendo novas possibilidades para o rastreamento populacional de doenças, educação em saúde em larga escala e monitoramento contínuo de grupos de risco. Essas soluções tecnológicas representam uma convergência entre as ciências da computação

e as necessidades clínicas, demonstrando potencial para superar limitações estruturais dos sistemas de saúde, especialmente em contextos de atenção primária caracterizados por recursos limitados e alta demanda assistencial (Topol, 2019; Meskó; Görög, 2020).

A arquitetura contemporânea desses sistemas baseia-se em modelos de LLMs combinados com *frameworks* de diálogo estruturado, permitindo não apenas a compreensão contextualizada da linguagem natural, mas também a adaptação dinâmica do fluxo conversacional conforme o perfil epidemiológico do usuário. Essa capacidade técnica é essencial para o rastreamento de condições crônicas de alta prevalência, como diabetes *mellitus*, hipertensão arterial, neoplasias e infecções sexualmente transmissíveis, onde os agentes conversacionais podem executar múltiplas funções: desde a aplicação automatizada de instrumentos de triagem validados até a disseminação de informações personalizadas sobre prevenção e autocuidado (Laranjo *et al.*, 2018; Kurniawan *et al.*, 2024).

O CB, entretanto, apresenta um cenário paradoxal: apesar de sua alta incidência (estimativa de 15.190 novos casos no Brasil para 2024, segundo o INCA) e da existência de fatores de risco estabelecidos (tabagismo, etilismo e exposição solar crônica), as iniciativas de rastreamento digital permanecem incipientes. Esta lacuna é preocupante considerando que aproximadamente 70% dos casos são diagnosticados em estágios avançados (T3 ou T4), quando as taxas de sobrevida em 5 anos não ultrapassam 40% (Santos *et al.*, 2023). As barreiras ao diagnóstico precoce são multifatoriais, incluindo desde a baixa percepção de risco entre a população até a distribuição desigual de profissionais especializados - no Brasil, a razão cirurgião-dentista/habitante nas regiões Norte e Nordeste é três vezes menor que no Sudeste.

Neste contexto, a implementação de um *chatbot* especializado em CB poderia atuar em três frentes: (1) como instrumento de triagem populacional, identificando indivíduos com perfis de risco elevado; (2) como ferramenta educativa, desmistificando conceitos e estimulando comportamentos preventivos; e (3) como sistema de referência inteligente, priorizando casos suspeitos para avaliação presencial, conforme critérios clínicos pré-definidos. Estudos de usabilidade demonstram que a aceitação dessas tecnologias entre populações vulneráveis está diretamente associada a três fatores: simplicidade da interface, adaptação linguística ao perfil sociocultural do usuário e integração transparente com os serviços locais de saúde (Palumbo; Nicola; Adinolfi, 2022).

A adoção de uma arquitetura multiagente em sistemas baseados em LLMs tem se consolidado como uma abordagem eficaz para equilibrar flexibilidade conversacional e controle semântico em aplicações de saúde digital. Nesse paradigma, diferentes agentes assumem papéis complementares, permitindo que o processo de interação seja dividido entre o planejamento do diálogo, a geração linguística contextualizada e a interpretação semântica das respostas. O uso de *frameworks*, como o LangChain, favorece a composição desses agentes em cadeias de processamento independentes e reutilizáveis, cada uma

responsável por uma etapa do raciocínio conversacional, por exemplo, a formulação de perguntas, o controle do fluxo de entrevista e a extração de valores clínicos. Essa integração torna-se relevante quando associada ao padrão HL7® FHIR®, que garante a representação estruturada e interoperável das informações coletadas, viabilizando a conversão de dados em linguagem natural em registros clínicos computáveis, auditáveis e reutilizáveis. A junção entre LLMs, *LangChain* e FHIR, apoiada por bancos de dados orientados a documentos, garante a rastreabilidade, persistência e escalabilidade das interações, além de permitir a evolução contínua da solução em ecossistemas complexos. Dessa forma, a arquitetura multiagente aplicada a LLMs representa uma evolução dos *chatbots* tradicionais, contemplando camadas de especialização e controle que fortalecem a confiabilidade clínica, a transparência operacional e a sustentabilidade técnica das aplicações em saúde digital (Laranjo *et al.*, 2018; Kurniawan *et al.*, 2024).

Assim, a interoperabilidade é garantida através da adoção do padrão FHIR, com mapeamento direto entre os elementos da conversa e recursos padronizados, por exemplo, o *Patient* (dados demográficos), *Observation* (sinais e sintomas) e *Condition* (histórico patológico). Com isso, viabiliza-se não apenas o armazenamento estruturado das informações, mas também a sua reutilização para fins secundários, desde a vigilância epidemiológica até o treinamento de modelos preditivos. O uso do servidor HAPI FHIR como *backbone* da solução assegura conformidade com as diretrizes internacionais de interoperabilidade enquanto facilita a integração com outras plataformas do ecossistema de saúde digital brasileiro (Hussain; Langer; Kohli, 2018).

2.4 *Frameworks* e ferramentas para implementação da solução: arquitetura técnica e fundamentação

A implementação de um sistema inteligente para rastreamento de doenças, baseado em agentes conversacionais, demanda uma arquitetura que integre PLN, armazenamento flexível de dados, interoperabilidade em saúde e comunicação eficiente com usuários finais. A solução proposta neste estudo adotará um conjunto de *frameworks* e ferramentas, alinhadas aos princípios da engenharia de software orientada a serviços (SOA) e da saúde digital, visando garantir escalabilidade, segurança e conformidade com padrões internacionais (Oshin; Campos, 2025; Schmiedmayer *et al.*, 2024; HL7 International, 2023).

2.4.1 *LangChain*: framework para construção de agentes inteligentes

O *LangChain* é um framework de código aberto projetado para facilitar o desenvolvimento de aplicações baseadas em modelos de linguagem, permitindo a criação de agentes autônomos capazes de interagir dinamicamente com múltiplas fontes de dados, planejar sequências de ações e manter contexto conversacional. A sua arquitetura modular viabiliza a composição de fluxos complexos, essenciais para sistemas de saúde, onde a precisão e a

rastreabilidade são críticas (Oshin; Campos, 2025).

No contexto de sistemas conversacionais inteligentes, o uso do *LangChain* como *framework* de orquestração tem permitido estruturar arquiteturas baseadas em LLMs segundo o paradigma multiagente, no qual diferentes componentes desempenham funções complementares de raciocínio e interação. Essa solução favorece a separação entre as camadas de planejamento do diálogo, geração de linguagem e interpretação semântica, assegurando maior controle sobre a coerência e a qualidade das respostas (Li *et al.*, 2024a).

A modularidade do *LangChain* facilita a testabilidade e a evolução incremental do sistema, permitindo a incorporação futura de novos módulos, como integração com APIs de prontuários eletrônicos ou sistemas de apoio à decisão clínica (Workman *et al.*, 2024).

2.4.2 *OpenAI* API e modelo GPT-4o mini: motor de linguagem natural

O modelo GPT-4o Mini, disponibilizado pela API da *OpenAI*, constitui o motor central de processamento de linguagem natural adotado neste estudo. Trata-se de um LLM otimizado para aplicações que demandam baixo tempo de resposta e custo computacional reduzido, sem comprometer a qualidade semântica das respostas. A sua arquitetura multimodal e de alta eficiência permite o processamento de entradas textuais com compreensão contextual avançada, o que o torna adequado a cenários interativos de coleta e análise de informações clínicas (Briganti, 2024).

A integração do modelo ao *framework LangChain* amplia as suas capacidades operacionais, pois inclui os mecanismos de memória de contexto, permitindo que o sistema mantenha a coesão e a continuidade de diálogos prolongados, uma característica essencial na coleta de históricos clínicos narrativos. Além disso, o *LangChain* viabiliza a personalização de *prompts*, ajustando o tom, o vocabulário e a profundidade das respostas, conforme o perfil do interlocutor, seja um usuário leigo ou um profissional de saúde. Outro aspecto relevante é o controle de temperatura, parâmetro que regula o grau de criatividade versus a precisão nas respostas geradas, mitigando o risco de alucinações semânticas em contextos críticos (Briganti, 2024).

A escolha por uma API gerenciada, em vez de um modelo local, justifica-se pela infraestrutura escalável e atualizações contínuas de segurança e desempenho oferecidas pela *OpenAI*, reduzindo a carga operacional da equipe de desenvolvimento.

2.4.3 MongoDB: Armazenamento de Dados Clínicos e Conversacionais

O MongoDB é adotado como tecnologia de persistência de dados clínicos e conversacionais por se tratar de um sistema de gerenciamento de banco de dados *NoSQL* orientado a documentos, utilizado em aplicações que exigem flexibilidade estrutural, escalabilidade horizontal e alta performance. A sua arquitetura baseada em documentos BSON/JSON permite representar informações clínicas e interacionais de forma semies-

truturada, adaptando-se às variações típicas dos modelos de informação em saúde e aos diferentes contextos das interações conduzidas pelos agentes inteligentes (Sen; Mukherjee, 2023).

Entre as suas principais vantagens destacam-se: (a) o esquema dinâmico, que possibilita a evolução contínua dos modelos de dados sem necessidade de migrações complexas, característica essencial em sistemas que integram múltiplos perfis clínicos e fluxos de entrevista; (b) a indexação eficiente, que permite a execução de consultas rápidas sobre os históricos de interação e registros de sessão, facilitando a recuperação de contextos anteriores e a personalização das respostas; e (c) a integração nativa com estruturas JSON, o que assegura compatibilidade direta com o padrão HL7® FHIR®, utilizado para serialização e intercâmbio de recursos clínicosm(Khoshroudi; Safaei; Soleimanjahi, 2025).

Para otimizar o desempenho em ambientes de alta demanda, utiliza-se a biblioteca Motor, um *driver* assíncrono do MongoDB para *Python*, que permite a execução de operações de leitura e escrita de forma concorrente e não bloqueante. Essa solução garante baixa latência, alta disponibilidade e resiliência transacional, atributos indispensáveis para a sustentação de sistemas conversacionais clínicos em tempo real e integrados a plataformas interoperáveis de saúde digital (Sen; Mukherjee, 2023).

3 METODOLOGIA

Este capítulo descreve os procedimentos metodológicos adotados para o desenvolvimento e a validação da solução proposta, em consonância com os objetivos estabelecidos na pesquisa. Apresentam-se o tipo de estudo, as etapas de modelagem, implementação e avaliação do protótipo, bem como a arquitetura técnica utilizada para integrar agentes inteligentes baseados em LLMs ao padrão HL7® FHIR®.

3.1 Tipo de Pesquisa

Este estudo caracterizou-se como uma pesquisa aplicada, de natureza tecnológica e de desenvolvimento experimental, com abordagem computacional e empírica, voltada à validação funcional de uma solução baseada em inteligência artificial. A pesquisa uniu os fundamentos de engenharia de software, inteligência artificial aplicada e interoperabilidade em saúde, com foco na implementação de um *chatbot* inteligente baseado em LLMs.

A solução foi desenvolvida com base no padrão HL7® FHIR®, adotando uma arquitetura modular e multicanal que integra diferentes camadas de processamento e comunicação. O núcleo da aplicação foi implementado em *FastAPI*, responsável pela orquestração dos agentes inteligentes, gerenciamento de sessões, controle de fluxos conversacionais e integração com os demais componentes da solução. Essa camada intermediária conecta os serviços de inferência (LLMs via *LangChain*), os bancos de dados (MongoDB e HAPI FHIR) e as interfaces de interação, a interface *web*, desenvolvida em *Streamlit*, e o canal de mensageria *WhatsApp*, integrado por meio da *EvolutionAPI*, utilizando *webhooks* para comunicação assíncrona e em tempo real (Li *et al.*, 2024b).

Do ponto de vista ético, o estudo não se caracteriza como pesquisa envolvendo seres humanos, pois não há coleta, registro ou análise de dados pessoais, tampouco interação com participantes. Os testes realizados limitaram-se à execução controlada de cenários simulados, com entradas artificiais e mensagens de validação destinadas apenas a verificar o funcionamento do *chatbot* e sua conformidade com o modelo FHIR. Dessa forma, o projeto encontra-se dispensado de submissão ao Comitê de Ética em Pesquisa (CEP), conforme as diretrizes nacionais vigentes.

A metodologia foi estruturada em quatro etapas principais, em consonância com os objetivos específicos deste estudo (Figura 1). A primeira etapa envolveu a modelagem da informação em saúde segundo o padrão HL7® FHIR®, com a definição e a estruturação dos elementos de dados clínicos necessários à coleta e à interoperabilidade das informações. A segunda etapa correspondeu ao desenvolvimento dos agentes inteligentes baseados em LLMs integrados ao *framework LangChain*, compreendendo três componentes: o Agente de

Orquestração, responsável pelo controle do fluxo conversacional; o Agente de Perguntas, encarregado de gerar enunciados naturais e contextualizados; e o Agente de Interação, voltado à extração, normalização e validação das respostas do usuário.

A terceira etapa contemplou a implementação da interface conversacional, desenvolvida em *Streamlit* e integrada ao *WhatsApp* por meio da *EvolutionAPI*, com intermediação da *API FastAPI* e uso de *webhooks* para comunicação assíncrona e atualização em tempo real. Por fim, a quarta etapa consistiu na avaliação dos resultados, concentrando-se na verificação da coleta correta das respostas via interface conversacional, na geração dos objetos JSON estruturados e na persistência bem-sucedida dos recursos clínicos no servidor HAPI FHIR, assegurando a interoperabilidade e a consistência dos dados trocados entre os módulos da solução.

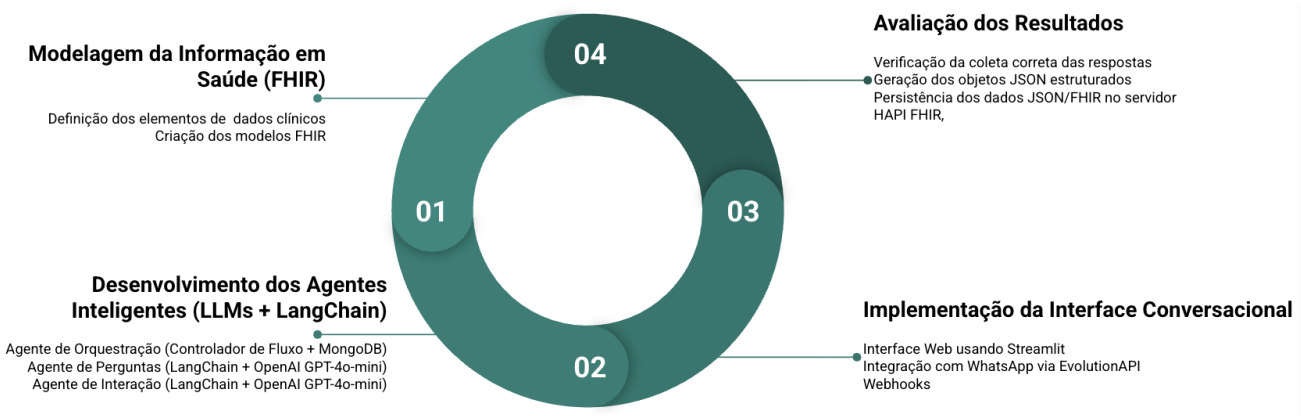

Figura 1 – Visão geral da metodologia.

Assim, neste estudo, adotou-se uma arquitetura modular, na qual o *FastAPI* desempenha o papel de núcleo de orquestração, integrando bancos de dados (MongoDB e HAPI FHIR), APIs de comunicação (*EvolutionAPI*) e servidores independentes de inferência e persistência, conforme detalhado nas seções seguintes.

3.2 Descrição das Etapas Metodológicas

3.2.1 Etapa 1 – Modelagem da informação em saúde estruturada em FHIR

Esta etapa consistiu na modelagem conceitual e técnica dos dados clínicos, alinhada ao padrão HL7® FHIR®. O processo iniciou-se com o levantamento e análise de diretrizes clínicas nacionais e internacionais, que orientaram a definição das variáveis (elementos de dados) necessárias para o rastreamento do CB. O escopo neste estudo foi restrito às variáveis relacionadas aos dados demográficos do paciente e os fatores de risco para o CB.

As variáveis foram mapeadas para os recursos específicos do FHIR, por exemplo *Patient*, *Observation*, *Condition* e *QuestionnaireResponse* (HL7 International, 2023). Após

o desenvolvimento dos modelos, foi realizada uma validação técnica utilizando o FHIR Validator.

Os procedimentos utilizados foram: levantamento de diretrizes clínicas; definição dos elementos de dados clínicos; mapeamento dos elementos de dados clínicos para os recursos FHIR; e validação técnica dos modelos usando o FHIR *Validator*.

3.2.2 Etapa 2 – Desenvolvimento dos Agentes Inteligentes

Nesta etapa foi desenvolvido o núcleo de inteligência conversacional, composto por múltiplos agentes especializados baseados em LLMs integrados ao *framework LangChain*, que processam dados clínicos em formato estruturado (FHIR).

A arquitetura foi composta por três agentes interdependentes:

- Agente de Orquestração: responsável por interpretar o modelo de informação (JSON/FHIR), gerenciar o estado da sessão e determinar o próximo campo a ser coletado, com base em regras determinísticas;
- Agente de Perguntas: encarregado de gerar perguntas curtas e contextualizadas a partir do histórico da conversa;
- Agente de Interação: dedicado à extração, normalização e validação das respostas, retornando valores em formato JSON estruturado, prontos para conversão FHIR.

Os agentes foram implementados em *Python* com *LangChain* e o modelo *OpenAI GPT-4o-mini*, com gerenciamento de estado persistido no MongoDB via *driver Motor (async)*. Essa camada garantiu memória contextual assíncrona, permitindo fluxos contínuos.

Todas as respostas processadas foram convertidas em recursos FHIR e persistidas no servidor HAPI FHIR, assegurando interoperabilidade e rastreabilidade.

Os procedimentos utilizados foram:

- Implementação de três agentes:
  - Agente de Orquestração: responsável por planejar e controlar o fluxo conversacional com base no modelo de informação FHIR, determinando a ordem de coleta dos campos obrigatórios e opcionais;
  - Agente de Perguntas: encarregado de gerar enunciados curtos e naturais, incorporando dicas de formato e opções de resposta conforme o tipo de dado esperado;
  - Agente de Interação: responsável por interpretar e normalizar as respostas dos usuários, aplicando validações automáticas e retornando os valores em formato JSON estruturado.

- Integração de memória contextual usando MongoDB, permitindo:
  - Armazenamento flexível de dados semiestruturados (JSON) referentes ao histórico de diálogo;
  - Consultas rápidas e recuperação eficiente de contextos anteriores;
  - Operações assíncronas via *Motor* (*Async Driver* para *Python*), garantindo escalabilidade e baixa latência.
- Aplicação de engenharia de prompts para controle e mitigação alucinações;
- Conversão automática das interações em recursos FHIR, com persistência no HAPI FHIR Server, assegurando interoperabilidade, rastreabilidade e consistência dos dados clínicos coletados.

3.2.3 Etapa 3 – Implementação da Interface Conversacional

A terceira etapa envolveu a implementação da interface de comunicação, desenvolvida em *Streamlit* e integrada ao *WhatsApp* via *EvolutionAPI*, utilizando *webhooks* para processamento de eventos em tempo real.

Essa camada foi responsável pela ingestão reativa de mensagens, validando *tokens* de segurança, distinguindo interações individuais de grupos e mantendo sessões de conversa associadas a cada número de telefone (*SESSION_BY_NUMBER*).

Além do canal de mensageria, a solução também contemplou uma interface *web* de testes e monitoramento para acompanhamento das sessões e artefatos FHIR.

Os procedimentos utilizados foram:

- Desenvolvimento da interface *web* utilizando *Streamlit*;
- Integração com *WhatsApp* via *EvolutionAPI* (*webhooks* para envio/recebimento em tempo real);
- Implementação de gestão de sessões com retomada automática de contexto;
- Configuração de *endpoint* para inicialização programática de conversas (coletas proativas).

3.2.4 Etapa 4 – Avaliação dos Resultados

A avaliação concentrou-se na verificação técnica e funcional do *pipeline* conversacional, validando o correto funcionamento do fluxo ponta a ponta, da coleta de respostas à persistência dos dados estruturados.

Foram verificados: (1) a coleta correta das respostas via interface conversacional (*web* e *WhatsApp*); (2) a geração automática dos objetos JSON estruturados; (3) a

persistência dos recursos FHIR no servidor HAPI FHIR, confirmando a interoperabilidade e a consistência dos dados.

Não foram conduzidas métricas de acurácia ou avaliação de desempenho quantitativo, uma vez que o objetivo principal foi a validação funcional do fluxo de dados e da arquitetura dos agentes.

3.3 Arquitetura Técnica da Solução

A arquitetura adotada foi modular, escalável e interoperável, composta por três camadas principais:

- Camada de Interação: responsável pela comunicação com o usuário via interface *web* (*Streamlit*) e *WhatsApp* (*EvolutionAPI*), processando mensagens por meio de *webhooks* assíncronos;
- Camada Lógica: compreende os agentes inteligentes (Orquestração, Perguntas e Interação), implementados com *LangChain* + *GPT-4o-mini*, responsáveis pelo fluxo conversacional e pela geração/validação de dados;
- Camada de Dados: integra o MongoDB (para armazenamento dos estados das sessões e dados semiestruturados) e o HAPI FHIR Server (para persistência dos recursos clínicos padronizados em FHIR).

Essa estrutura modular garantiu desacoplamento entre os serviços, escalabilidade horizontal e conformidade com os princípios da Estratégia de Saúde Digital para o Brasil (2020–2028) (Brasil, 2020), priorizando interoperabilidade, inovação e sustentabilidade técnica.

4 AVALIAÇÃO EXPERIMENTAL

Este capítulo apresenta a avaliação experimental da solução proposta, alinhada ao principal objetivo de desenvolver agentes inteligentes baseados em LLMs, integrados ao padrão HL7® FHIR®, capazes de orquestrar interações personalizadas e interoperáveis para coleta e estruturação de informações clínicas, com foco no rastreamento e monitoramento da população de risco para o CB. A análise concentrou-se em demonstrar como a modelagem FHIR, a arquitetura multiagente (Orquestração, Perguntas e Interação) e a infraestrutura de dados (MongoDB e HAPI FHIR) se integram para transformar linguagem natural em recursos clínicos computáveis e interoperáveis.

Complementarmente, a camada de orquestração da solução foi implementada em *FastAPI*, atuando como *gateway* de serviços entre os canais de interação (*Streamlit* e *WhatsApp/EvolutionAPI*), os agentes inteligentes (Orquestração, Perguntas e Interação) e a camada de dados (MongoDB e HAPI FHIR). Essa API concentrou os *endpoints* de sessão, *webhooks* e persistência, garantindo desacoplamento, controle transacional do fluxo e rastreabilidade ponta a ponta.

A seção inicia pela modelagem da informação em FHIR, detalhando o perfil do recurso *Patient* e as regras de validação adotadas. Em seguida, descreve-se o núcleo de agentes inteligentes e o seu papel na geração de perguntas, extração/normalização de respostas e controle determinístico do fluxo conversacional. Na sequência, apresenta-se a interface conversacional (*web/Streamlit* e *WhatsApp* via *EvolutionAPI*), com ênfase na ingestão reativa por *webhooks*, manutenção de estado e retomada de contexto. Por fim, reportam-se os achados sobre persistência e interoperabilidade (envio de recursos ao HAPI FHIR) e a avaliação funcional do *pipeline*, incluindo evidências de consistência semântica, completude dos campos e confiabilidade operacional em cenários simulados. Ao longo do capítulo, figuras e exemplos de mensagens/recursos ilustram os resultados empíricos e sustentam a discussão técnica.

4.1 Modelagem da informação em saúde em FHIR

A primeira etapa da pesquisa resultou no desenvolvimento de um Modelo de Informação (MI) para o recurso *Patient*, estruturado em conformidade com o padrão HL7® FHIR®. Esse modelo teve como finalidade capturar dados demográficos e informações administrativas de indivíduos pertencentes ao grupo de risco para o CB, contemplando homens acima de 50 anos, indivíduos com lesões bucais persistentes há mais de 15 dias e pacientes com fatores de risco como tabagismo e etilismo (Chamoli; al., 2021).

O processo iniciou-se com o levantamento e análise de diretrizes clínicas nacionais

e internacionais voltadas à prevenção e detecção precoce do CB. A partir desse referencial, foram definidos os elementos de dados clínicos essenciais, incluindo identificadores (CPF ou CNS), informações pessoais (nome, data de nascimento, sexo, nome da mãe), dados de contato (telefone, e-mail), endereço completo (com CEP, cidade, estado, bairro e complemento) e geolocalização (latitude e longitude). Também foram incluídos fatores de risco relacionados ao tabagismo, etilismo e presença de lesões suspeitas (Apêndice A).

Com base nessas variáveis (elementos de dados), procedeu-se ao mapeamento das informações clínicas usando o padrão HL7® FHIR®. O MI foi mapeado para os recursos/elementos de dados FHIR, destacando-se o *Patient* (dados sociodemográficos e de contato), *Location* (microrregião de saúde que o paciente reside) e *RiskAsessment* (para mapear os fatores de risco que o paciente possui) - Figura 2. Esse mapeamento garantiu a representação estruturada, padronizada e interoperável das informações (Schmiedmayer *et al.*, 2024; HL7 International, 2023).

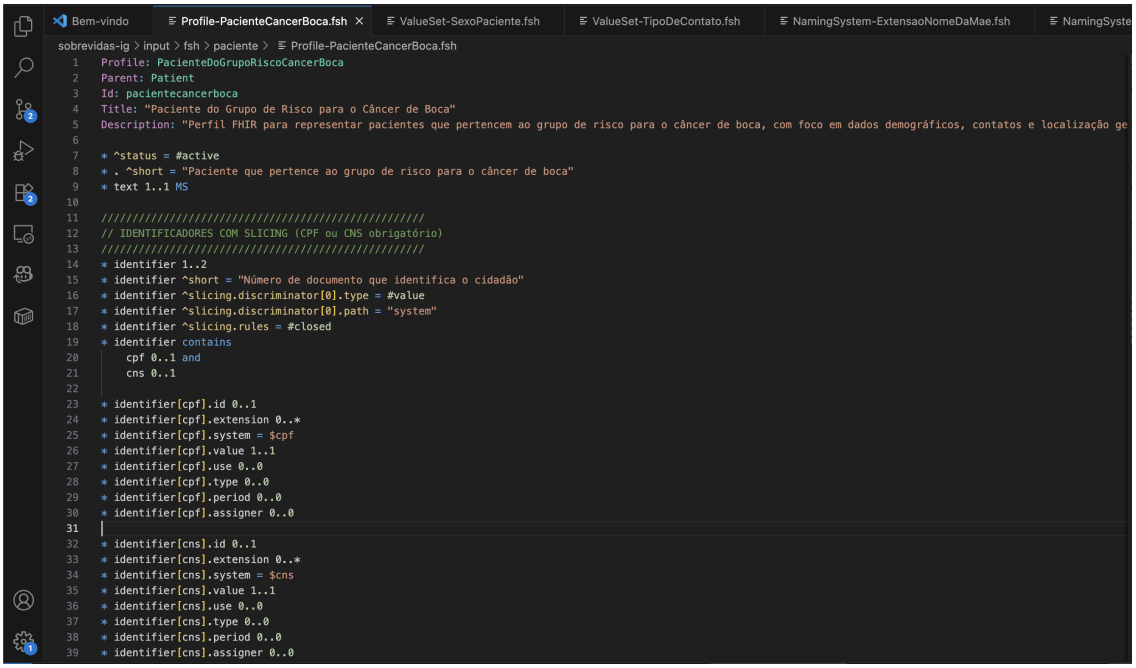

Figura 2 – Perfil FHIR em FSH para paciente do grupo de risco.

O modelo resultante, denominado Perfil FHIR "*Paciente do Grupo de Risco para o Câncer de Boca*" (FIgura 3), foi especificado em FHIR *Shorthand* (FSH). O perfil estabeleceu cardinalidades obrigatórias, extensões (por exemplo, nome da mãe e geolocalização do domicílio) e invariantes de validação para assegurar a integridade e qualidade dos dados. Entre os principais destaques:

- obrigatoriedade de CPF e/ou CNS como identificadores primários;
- exigência de nome completo, data de nascimento e sexo;

- pelo menos um meio de contato (telefone ou e-mail), com validação de formatos conforme padrões nacionais e internacionais (telefone em formato +55 DD XXXXX-XXXX, e-mail);

- registro completo, incluindo CEP validado (XXXXX-XXX), cidade, estado e bairro, além de extensão para latitude e longitude;

- fatores de risco, sendo os atributos específicos para caracterizar os pacientes como tabagistas, etilistas, maiores de 50 anos e portadores de lesões bucais suspeitas.

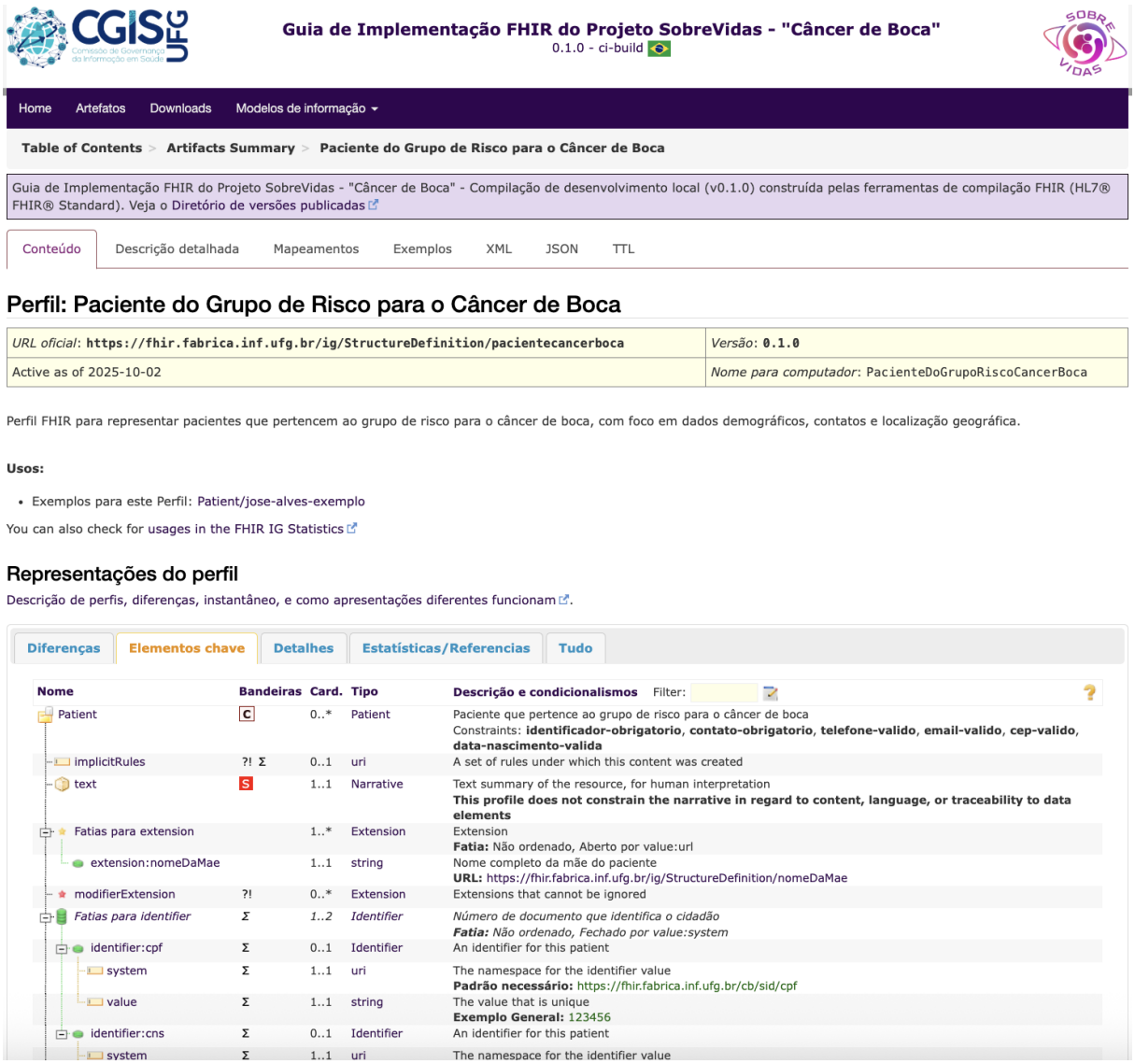

Figura 3 – Guia de implementação do paciente do grupo de risco.

Para garantir a aplicabilidade prática do modelo, foram elaborados cenários de uso simulados, representando situações reais de atenção primária à saúde:

Cenário 1 – Registro inicial de paciente com dados demográficos e contatos atualizados

A agente comunitária de saúde Luana realiza uma visita domiciliar no bairro Jardim das Palmeiras utilizando seu celular com acesso ao *WhatsApp*. Durante a visita, ela inicia uma conversa com o assistente virtual da Plataforma, que conduz a entrevista de forma simples e interativa. O assistente faz perguntas curtas, como "Qual é o nome completo do paciente?", "Qual a data de nascimento?" ou "Qual o telefone para contato?".

Enquanto Luana responde, a solução verifica automaticamente se os dados informados estão no formato correto (por exemplo, CPF com 11 dígitos, CEP válido e telefone com DDD). Ao preencher o endereço, a agente informa também a latitude e a longitude, permitindo o registro geográfico da residência do paciente para fins de monitoramento territorial. Ao final da coleta, a solução confirma o registro e armazena as informações do paciente na base de dados, estruturadas segundo o padrão internacional FHIR.

Cenário 2 – Atualização cadastral de paciente já conhecido pela unidade

Na recepção da Unidade Básica de Saúde do Setor Sul, Juliana identifica que o telefone do paciente Carlos Martins (58 anos) está desatualizado. Pelo computador da unidade, ela acessa a interface *web* da Plataforma e localiza o cadastro digitando o CPF ou o número do CNS. O assistente virtual auxilia na verificação dos dados, sugerindo correções quando há erros de digitação, por exemplo, se o e-mail estiver incompleto ou o número de telefone for inválido.

Juliana atualiza as informações e salva o registro, que é automaticamente sincronizado com o banco de dados da plataforma, garantindo que o cadastro do paciente permaneça completo e atualizado. Essa operação é voltada exclusivamente à manutenção administrativa dos dados de identificação e contato, sem envolver o registro de informações clínicas.

Cenário 3 – Registro de novo paciente sem CPF, identificado apenas pelo CNS

Durante um mutirão de cidadania, Marcela utiliza o WhatsApp da Plataforma para cadastrar Francisco dos Santos (61 anos), que não possui CPF ativo, mas apresenta o Cartão Nacional de Saúde (CNS). O assistente virtual reconhece a situação e orienta Marcela a utilizar o CNS como identificador principal do paciente.

Em seguida, conduz a entrevista de forma dialogada, solicitando nome completo, data de nascimento, nome da mãe, sexo, endereço, telefone e e-mail. A solução valida cada informação conforme é digitada e, ao final, confirma que o cadastro foi realizado com sucesso, garantindo que os dados estejam armazenados de forma segura e padronizada.

A validação técnica do perfil foi realizada utilizando o FHIR *Validator* (Figura 4),

visando garantir a conformidade com a especificação FHIR R4 - JSON.

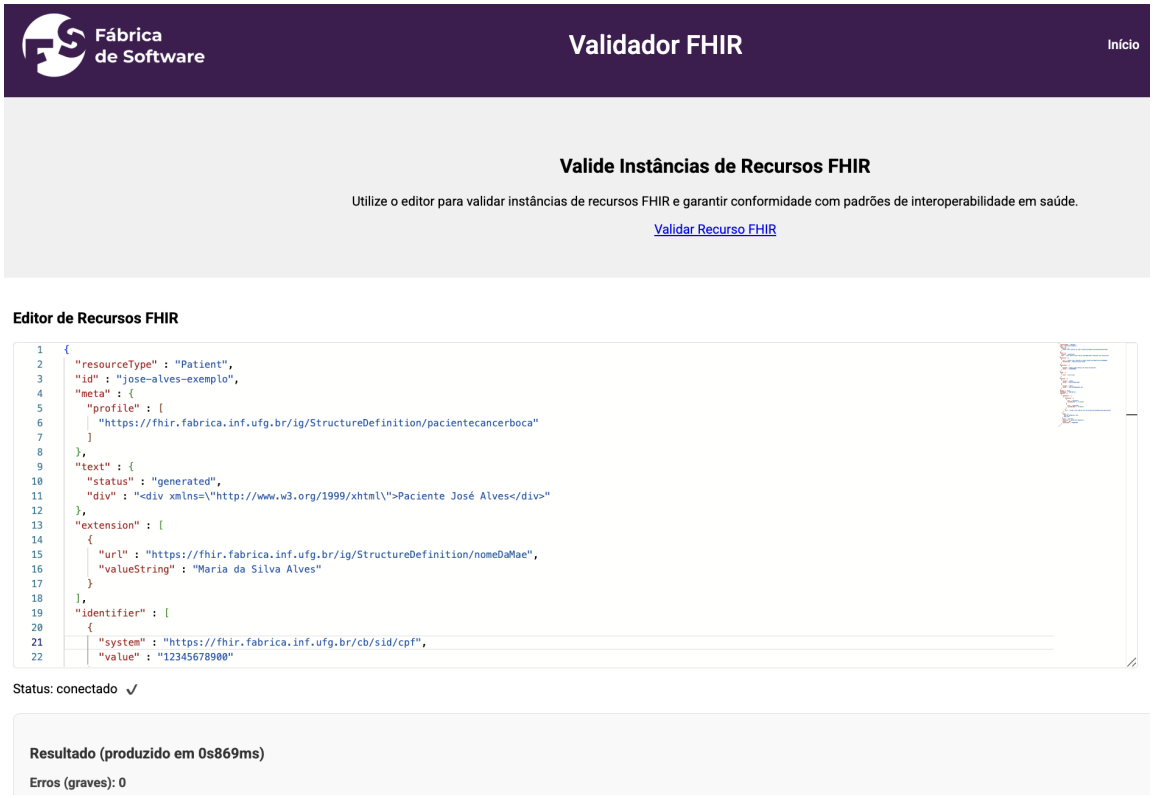

Figura 4 – Validação técnica no FHIR *Validator*.

Abaixo, um exemplo de *Patient* em JSON.

```
{
 "resourceType" : "Patient",
 "id" : "jose-alves-exemplo",
 "meta" : {
   "profile" : [
     "https://fhir.fabrica.inf.ufg.br/ig/StructureDefinition/pacientecancerboca"
   ]
 },
 "text" : {
   "status" : "generated",
   "div" : "<div xmlns=\"http://www.w3.org/1999/xhtml\">Paciente José Alves</div>"
 },
 "extension" : [
   {
     "url" : "https://fhir.fabrica.inf.ufg.br/ig/StructureDefinition/nomeDaMae",
     "valueString" : "Maria da Silva Alves"
```

```
    }
  ],
  "identifier" : [
    {
      "system" : "https://fhir.fabrica.inf.ufg.br/cb/sid/cpf",
      "value" : "12345678900"
    }
  ],
  "name" : [
    {
      "text" : "José Alves"
    }
  ],
  "telecom" : [
    {
      "system" : "phone",
      "value" : "+55 62 98765-4321"
    },
    {
      "system" : "email",
      "value" : "jose.alves@exemplo.com"
    }
  ],
  "gender" : "male",
  "birthDate" : "1963-05-15",
  "address" : [
    {
      "extension" : [
        {
          "extension" : [
            {
              "url" : "latitude",
              "valueDecimal" : -16.123456
            },
            {
              "url" : "longitude",
              "valueDecimal" : -48.654321
            }
          ],
```

```
          "url" : "https://fhir.fabrica.inf.ufg.br/ig/StructureDefinition
          /geolocation"
        }
      ],
      "line" : [
        "Rua das Palmeiras, 123",
        "Apto 101"
      ],
      "city" : "Goiânia",
      "district" : "Jardim das Palmeiras",
      "state" : "GO",
      "postalCode" : "74000-000"
    }
 ]
}
```

4.2 Agentes Inteligentes

Nesta etapa foi desenvolvido o núcleo de inteligência conversacional, estruturado em uma arquitetura multiagente capaz de transformar dados não estruturados em informações clínicas padronizadas no formato HL7® FHIR®. O desenvolvimento foi realizado em *Python*, integrando o *framework LangChain* ao modelo *OpenAI GPT-4o-mini*, com persistência assíncrona de estado no MongoDB (via *driver Motor*). Essa composição permitiu o funcionamento contínuo de sessões conversacionais e a integração direta com o HAPI FHIR *Server* para validação e armazenamento dos recursos gerados (Schmiedmayer *et al.*, 2024).

Toda a comunicação entre os agentes e os demais serviços foi mediada por uma API *FastAPI*, responsável por expor *endpoints* de criação/gestão de sessões, encaminhar tarefas aos agentes e consolidar as respostas normalizadas antes do mapeamento para FHIR.

A arquitetura multiagente foi composta por três componentes interdependentes: o Agente de Orquestração (*OrchestratorAgent*) foi responsável por interpretar o modelo de informação (JSON/FHIR), gerenciar o estado da sessão persistido no MongoDB e determinar o próximo campo a ser coletado, com base em regras determinísticas. Esse agente controla o fluxo da conversa e garante a completude do questionário clínico. O Agente de Perguntas (*QuestionAgent*) gerou enunciados curtos e naturais para cada campo, incorporando opções e dicas de formato conforme o tipo de dado, a partir do histórico recente da conversa. Já o Agente de Interação (*InteractionAgent*) conduziu a extração e a normalização dos valores informados pelo usuário, aplicando validações automáticas e retornando respostas em formato JSON estruturado (Tabela 2) - disponível no *GitHub*.

Tabela 2 – Agentes inteligentes, funções, tecnologias e saídas principais.

Agente	**Função principal**	**Tecnologias (do código)**	**Saída principal**
Orchestrator Agent	Decide o próximo campo (regra obrigatórios→opcionais) e pede ao *QuestionAgent* a pergunta.	Python (regras), *InterviewPlan / SessionState*.	*next_slot*, *question*, *slot_spec*.
Question Agent	Gera perguntas curtas e naturais para cada *slot*, com opções e dica de formato, usando histórico recente.	LangChain, *ChatOpenAI (gpt-4o-mini)*, *ChatPromptTemplate*, *StrOutputParser*; *fallback* determinístico quando sem API.	Texto da pergunta.
Interaction Agent	Extrai e normaliza respostas; faz *post-processing* (data/telefone), valida e retorna JSON.	*LangChain*, *ChatOpenAI (gpt-4o-mini)*, *parser* de JSON, *validate_value*.	*JSON { value, normalized, confidence }* + *flags*.

Na camada de orquestração, a *FastAPI* disponibilizou serviços como *POST /conversation/sessions* (inicialização de sessão), *POST /conversation/answer* (entrega de mensagens ao *InteractionAgent*) e *POST /conversation/finalize* (disparo do mapeamento e envio ao HAPI FHIR), permitindo instrumentação e auditoria do ciclo conversacional.

Esses agentes operaram de forma colaborativa, compondo um *pipeline* inteligente que inicia com a geração de perguntas adaptativas, prossegue com a extração semântica das respostas, e termina na conversão automática para os recursos FHIR. A Figura 5 ilustra a arquitetura geral da solução, destacando as camadas de ingestão, processamento e persistência.

Essa arquitetura tecnológica, concebida para sustentar o núcleo inteligente de coleta e processamento de dados clínicos, foi projetada para atender aos desafios de velocidade, variedade e valor inerentes a ambientes de *Big Data* em saúde, oferecendo uma infraestrutura escalável e resiliente. Ela foi desenvolvida com base em conteneirização por meio do *Docker* e orquestração via *Docker Compose*, permitindo a modularização dos serviços e o isolamento funcional de cada componente (O'Connor *et al.*, 2017). O conjunto de serviços foi organizado em três camadas principais, que refletem o fluxo lógico do processamento de dados, desde a ingestão até a persistência estruturada:

- Camada de Ingestão e Interação (*EvolutionAPI*, UI): composta por componentes de borda (*edge components*) responsáveis pela recepção e roteamento dos dados provenientes das interações em tempo real. Essa camada suporta múltiplos canais de comunicação, como o *WhatsApp* (via *webhooks*) e a interface *web* interativa desenvolvida em *Streamlit*, garantindo responsividade e integração fluida com os usuários.

- Núcleo de Processamento (API): constitui o núcleo da solução, contemplando a lógica de negócio e os agentes de inteligência artificial baseados em LLMs integrados ao *framework LangChain*. Essa camada é *stateless*, ou seja, não mantém estado local, o que permite escalabilidade horizontal e balanceamento dinâmico de carga. O núcleo orquestra as interações entre usuários, agentes e bancos de dados, assegurando consistência transacional entre as mensagens e os objetos clínicos gerados. A implementação foi realizada em *FastAPI*, que atuou como *gateway* de comunicação entre os canais de interação (*Streamlit* e *WhatsApp/EvolutionAPI*), os agentes de orquestração e inferência, e os repositórios de dados (MongoDB e HAPI FHIR). Além de expor *endpoints* REST para criação e gerenciamento de sessões, a *FastAPI* forneceu uma interface interativa de documentação e teste via *Swagger UI* (Figura 6), permitindo a inspeção e execução controlada das rotas, o que facilitou a validação funcional e a rastreabilidade das operações da solução.

- Camada de Persistência Poliglota: implementada para otimizar o armazenamento de diferentes tipos de dados, conforme sua estrutura e propósito. O MongoDB foi utilizado como um *datalake* operacional, destinado a armazenar o estado das sessões conversacionais e o histórico de interações. A sua capacidade de lidar com dados semiestruturados (JSON) mostrou-se essencial para persistir informações dinâmicas e heterogêneas oriundas dos diálogos. Já o HAPI FHIR *Server* foi empregado como *data warehouse* clínico, responsável pela persistência dos dados estruturados no formato HL7® FHIR®, assegurando interoperabilidade semântica e rastreabilidade completa do dado clínico.

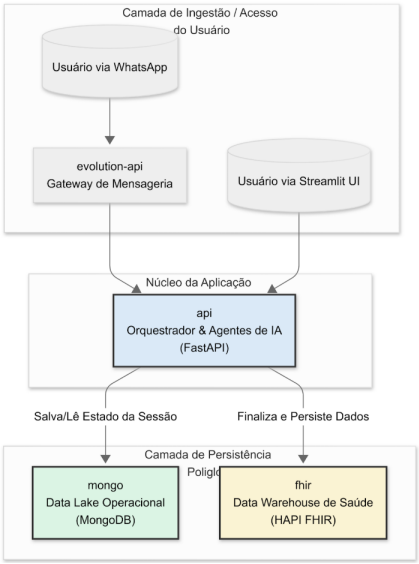

Figura 5 – Diagrama da Arquitetura implementada, ilustrando o desacoplamento dos serviços e o fluxo de dados desde a ingestão até a camada de persistência estruturada. A *FastAPI* centraliza a orquestração entre canais, agentes e persistência FHIR.

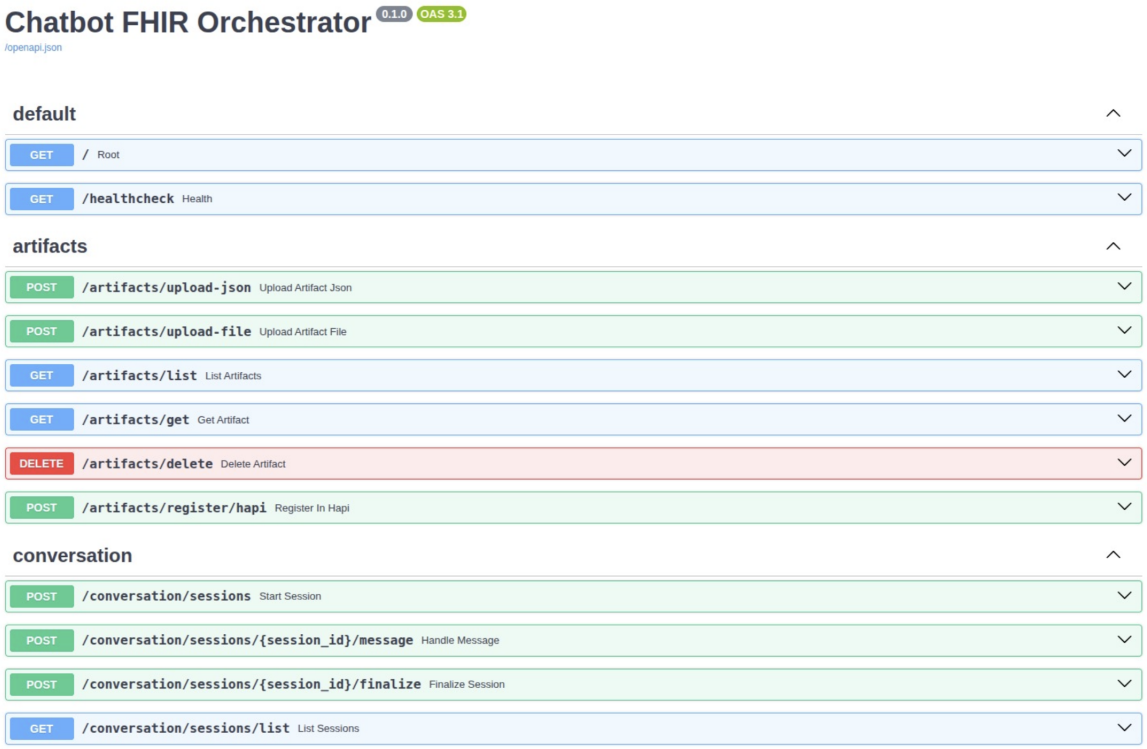

Figura 6 – Interface de documentação automática da API *FastAPI* (*Swagger* UI) exibindo os endpoints de artefatos e sessões do *Chatbot FHIR Orchestrator*.

Portanto, o núcleo da solução é um *pipeline* de processamento inteligente que transforma texto livre em recursos clínicos FHIR por meio de uma arquitetura multiagente (Orquestração, Perguntas e Interação) apoiada por LLMs (*GPT-4o-mini*) integrados ao *LangChain*. O encadeamento compreende: ingestão reativa, gerenciamento de estado, geração de perguntas contextuais, extração/normalização e persistência FHIR.

Em relação às interações móveis, o ponto de partida é o *webhook* que recebe o evento *messages.upsert* (novas mensagens) e filtra ruído operacional (eventos de *status*/presença e conversas em grupo), priorizando sessões individuais. Cada requisição é autenticada por *token* e, a partir do número do remetente, a solução mantém uma sessão ativa associada a um *session_id* único (*SESSION_BY_NUMBER*), com retomada de contexto entre mensagens e dias. A ingestão é assíncrona e orientada a eventos, desacoplada do núcleo de IA, o que contribui para absorver picos de tráfego sem degradar o tempo de resposta (Figura 7).

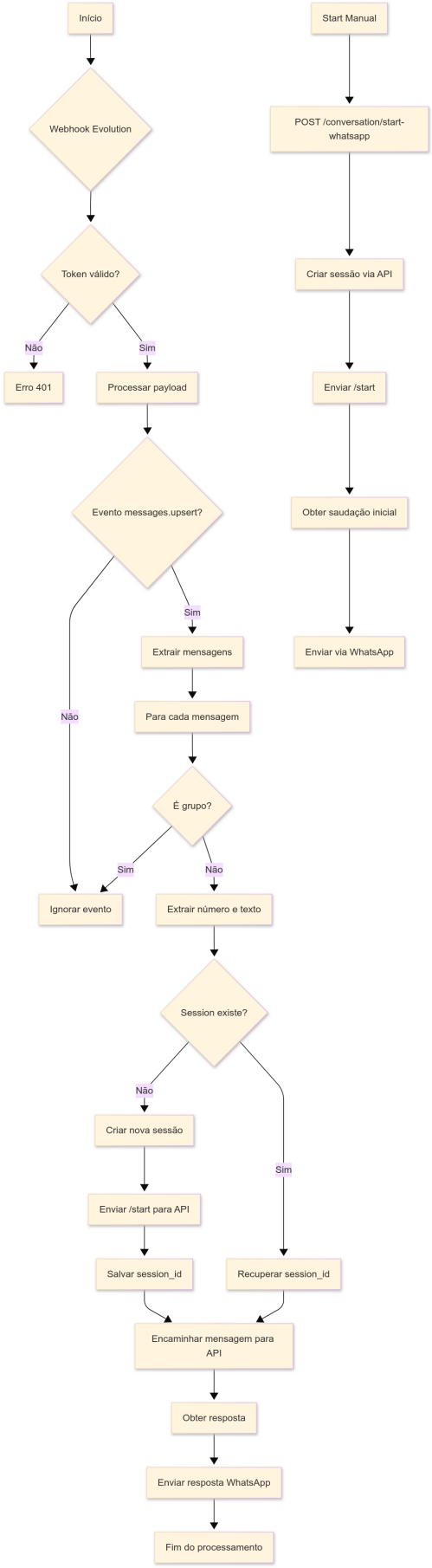

Figura 7 – Fluxo de ingestãoestado via WhatsApp e dados coletados.

Essa configuração modular possibilitou a comprovação empírica da viabilidade técnica da arquitetura proposta, demonstrando sua capacidade de integração entre agentes inteligentes, banco de dados e servidores FHIR. Além disso, o desacoplamento entre os módulos garantiu resiliência e manutenção independente dos componentes, atendendo aos princípios de escalabilidade e sustentabilidade tecnológica (Li *et al.*, 2024b; Kurniawan *et al.*, 2024).

4.2.1 Agente de Orquestração

A orquestração conversacional constituiu o núcleo lógico da solução inteligente, responsável por conduzir a coleta de informações de forma estruturada e determinística, garantindo completude e consistência dos dados. Essa camada foi implementada pelo *OrchestratorAgent*, em colaboração com o *QuestionAgent*, compondo um ciclo de interação entre controle de fluxo e geração de linguagem natural.

O *OrchestratorAgent* operou como um gerenciador de estado, encarregado de interpretar o modelo de informação (representado em JSON/FHIR), identificar os campos pendentes de preenchimento e determinar a sequência das perguntas a serem apresentadas ao usuário. A política de coleta foi definida por um plano de entrevista (*InterviewPlan*), que priorizava a obtenção de todos os campos obrigatórios antes de avançar para os opcionais, uma estratégia determinística que assegurou a completude mínima dos recursos clínicos exigidos pelo perfil FHIR.

A cada iteração do diálogo, o agente verificava o estado da sessão (*SessionState*), mapeando os campos já preenchidos e aqueles ainda em aberto. Quando identificado um novo campo pendente, o *OrchestratorAgent* acionava o *QuestionAgent*, responsável por formular o enunciado em linguagem natural. Esse agente utilizava o modelo *OpenAI GPT-4o-mini*, via *LangChain*, para gerar perguntas curtas, contextualizadas e adaptadas ao histórico recente da conversa, evitando repetições e mantendo uma comunicação fluida e humanizada. Em situações de ausência de contexto ou falha de rede, a solução aplicava uma política de fallback, utilizando perguntas padrão predefinidas no modelo de entrevista.

O ciclo de orquestração e geração de linguagem natural refere-se à colaboração entre os dois agentes e o mecanismo de transição entre os estados de coleta, resposta e validação. Essa separação entre lógica de fluxo (orquestração) e geração de linguagem ampliou a manutenibilidade da solução, permitindo a substituição ou atualização de modelos de linguagem sem comprometer o núcleo determinístico da coleta.

Do ponto de vista técnico, o *OrchestratorAgent* foi implementado em *Python* e integrou os módulos *InterviewPlan*, *SessionState* e *SlotSpec*, responsáveis por armazenar a especificação dos campos, o estado da sessão e a tipificação dos dados esperados. O algoritmo de priorização segue uma abordagem sequencial e determinística, conforme ilustrado no Trecho de Código a seguir, em que o agente percorre a lista de campos

definidos no modelo, selecionando primeiro os obrigatórios não preenchidos e, em seguida, os opcionais.

```
# app/src/agents/orchestrator_agent.py
class OrchestratorAgent(AgentInterface):
    def next_pending_slot(self, state: SessionState) -> Optional[SlotSpec]:
        # 1) obrigatórios não preenchidos
        for s in self.plan.slots:
            if s.required and s.slot_id not in state.filled:
                return s
        # 2) opcionais não preenchidos
        for s in self.plan.slots:
            if not s.required and s.slot_id not in state.filled:
                return s
        return None

    async def execute(self, state: SessionState) -> dict:
        slot = self.next_pending_slot(state)
        if not slot:
            return {"done": True, "message": "Coleta concluída.", "next_slot":
            None}
        question = slot.prompt_template or f"Informe {slot.label or
        slot.slot_id}:"
        return {"done": False, "next_slot": slot.slot_id, "question": question,
        "slot_spec": slot.dict()}
```

Esse mecanismo assegurou previsibilidade e rastreabilidade no processo de coleta, uma vez que cada *slot* representava um campo específico do modelo FHIR (por exemplo, identificador, nome, data de nascimento, contato, endereço). O encerramento do fluxo ocorria apenas quando não restavam *slots* pendentes, momento em que o agente retornava o marcador de conclusão da sessão.

Além de garantir a consistência lógica, o agente também desempenhou papel central na integridade dos dados clínicos coletados, assegurando que a estrutura do questionário seguisse as regras de cardinalidade e obrigatoriedade definidas nos perfis FHIR. Essa estratégia eliminou omissões comuns em coletas manuais e reduziu a necessidade de retrabalho na etapa de persistência (Li *et al.*, 2024a).

A implementação conjunta dos agentes de orquestração e de geração de linguagem natural demonstrou eficiência no gerenciamento do diálogo e robustez na coleta de dados, validando o modelo proposto de controle determinístico aliado à flexibilidade semântica

dos LLMs (Schmiedmayer *et al.*, 2024). Essa arquitetura híbrida, composta por regras explícitas de controle e componentes probabilísticos de linguagem, resultou em uma solução capaz de conduzir entrevistas clínicas automatizadas com qualidade técnica e coerência semântica, mantendo a conformidade com o padrão HL7® FHIR® (HL7 International, 2023).

4.2.2 Agente de Interação

O Agente de Interação (*InteractionAgent*) constitui o núcleo de processamento linguístico e semântico da solução, responsável pela extração, normalização e validação de dados clínicos informados pelos usuários em linguagem natural. Este agente foi desenvolvido em *Python*, utilizando o *framework LangChain* e o modelo *OpenAI GPT-4o-mini*, com temperatura configurada em 0, garantindo comportamento determinístico e reduzindo a variabilidade das respostas.

O processo de extração é conduzido por meio de engenharia de *prompts* estruturada, que orienta o modelo de linguagem a responder exclusivamente em formato JSON, obedecendo a um contrato rígido de saída. Cada iteração segue uma sequência composta por três estágios principais:

Engenharia de *prompt* e contextualização: para cada campo (*slot*) definido no modelo de informação FHIR, o agente monta dinamicamente um prompt com as instruções necessárias para guiar o LLM. O prompt inclui o rótulo do campo, o tipo de dado esperado e exemplos de valores válidos. O modelo é instruído a retornar um objeto JSON no formato:

```
{
"value": <valor_bruto>, "normalized": <valor_normalizado_ou_nulo>,
"confidence": 0.0-1.0
}
```

Essa estrutura funciona como um contrato semântico, garantindo a previsibilidade da resposta e facilitando a posterior conversão para recursos FHIR.

Inferência e Normalização Automática: O modelo executa o processo de inferência a partir da entrada do usuário, aplicando transformações semânticas diretamente no *prompt*. Regras explícitas de normalização foram incorporadas, tais como: datas no formato padrão ISO (AAAA-MM-DD); telefones com código internacional e local (DDI+DDD), adicionando +55 para números brasileiros; mapeamento de gênero administrativo para códigos FHIR (Feminino→*female*, Masculino→*male*, Outro→*other*).

A resposta do modelo é validada localmente por uma função de *parsing* (*_safe_json_loads*), capaz de recuperar a estrutura JSON mesmo em saídas parcialmente formatadas, mitigando falhas de interpretação do LLM.

Validação e Pós-Processamento de Domínio: Após a inferência, os valores retornados passam por uma etapa de validação sintática e semântica conduzida pela função *validate_value()*. Essa função aplica expressões regulares e regras de negócio alinhadas aos invariantes dos perfis FHIR, como formato de telefone, e-mail, CEP, e consistência de datas de nascimento.

Caso algum valor viole as regras de domínio, o agente sinaliza *needs_clarification = true*, armazena mensagens de aviso específicas e gera uma pergunta de *follow-up* orientando o usuário à correção do campo.

Quando o modelo retorna um JSON inválido, aplica-se um mecanismo de *fallback*, preservando o valor bruto, atribuindo normalized = null e confiança baixa, de forma a impedir a persistência de dados inconsistentes.

O trecho de código abaixo ilustra a estrutura central do Agente de Interação:

```
    # app/src/agents/interaction_agent.py
SYSTEM_PROMPT = """
Você é um assistente que extrai e NORMALIZA valores para um campo (slot) FHIR do
paciente.
- Deve responder SOMENTE com JSON no seguinte formato:
  {"value": <valor_bruto>, "normalized": <valor_normalizado_ou_nulo>,
  "confidence": 0.0-1.0}
- Se estiver incerto, deixe "normalized" nulo e "confidence" baixa.
- Para 'date': output AAAA-MM-DD.
- Para 'phone': inclua DDI+DDD se ausente (+55 se usuário no Brasil).
- Para 'gender' (code): mapear {Feminino->"female", Masculino->"male",
Outro->"other"}.
- Não narre nada fora do JSON.
"""

class InteractionAgent(AgentInterface):
    def __init__(self, model: str = "gpt-4o-mini"):
        self.llm = ChatOpenAI(model=model, api_key=os.getenv("OPENAI_API_KEY"),
        temperature=0)

    async def extract(self, slot: SlotSpec, user_text: str) -> SlotAnswer:
        user_prompt = f"""
Campo: {slot.label or slot.slot_id}
Tipo: {slot.type}
Exemplos: {slot.examples}
```

```
Entrada do usuário: \"{user_text}\"
Responda SOMENTE o JSON pedido.
"""
        prompt = ChatPromptTemplate.from_messages([("system", SYSTEM_PROMPT),
        ("human", user_prompt)])
        chain = prompt | self.llm | StrOutputParser()
        raw = await chain.ainvoke({})
        ...
```

Durante a execução, o agente instancia um objeto *ChatOpenAI* com o parâmetro *temperature=0*, promovendo consistência e reprodutibilidade das respostas, aspecto essencial em aplicações clínicas. O *pipeline* completo (*prompt* → LLM → *StrOutputParser* → validação) une a flexibilidade da inferência neural com a segurança de mecanismos determinísticos locais.

Além disso, o *InteractionAgent* foi projetado para interoperar de forma assíncrona com o MongoDB, garantindo armazenamento eficiente de estados conversacionais e rastreabilidade completa do histórico de coleta. Cada resposta validada é consolidada como um recurso FHIR e enviada ao HAPI FHIR *Server*, assegurando interoperabilidade e conformidade técnica (Figura 8).

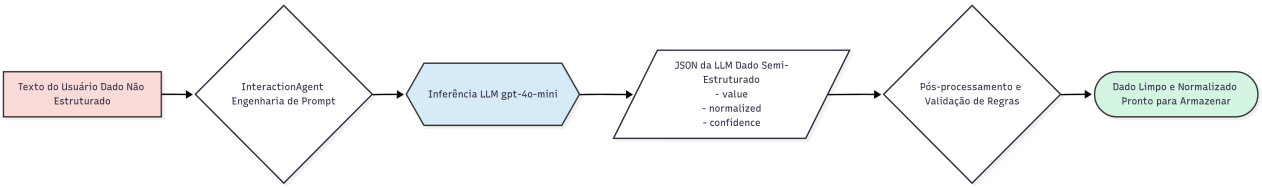

Figura 8 – Fluxo de processamento do *InteractionAgent*, demonstrando o uso do modelo para extração e normalização de dados clínicos em formato FHIR.

Do ponto de vista técnico, este agente aborda diretamente o desafio da Variedade dos dados, transformando a linguagem livre dos usuários em registros clínicos computáveis, interoperáveis e semanticamente corretos. Essa capacidade é essencial para o rastreamento automatizado de populações de risco, permitindo a geração de bases de dados estruturadas e reutilizáveis para vigilância epidemiológica e integração com sistemas nacionais de saúde (Li *et al.*, 2024b).

4.2.3 Agente de Perguntas

O *QuestionAgent* constitui a camada de geração de linguagem natural responsável por converter especificações de campos clínicos (*slots*) em perguntas curtas, claras e contextuais, assegurando coesão discursiva e redução de ambiguidade. Diferentemente

do controle de fluxo (agente de orquestração), o *QuestionAgent* foca no enunciado ideal para elicitar cada dado requerido, operando sobre três princípios: (i) contextualização por histórico recente, (ii) sinalização explícita de formato quando pertinente e (iii) oferta direta de opções quando há *ValueSets* conhecidos.

Do ponto de vista operacional, o agente utiliza *LangChain + ChatOpenAI (GPT-4o-mini)* com *temperature = 0,2*, um ajuste que busca equilíbrio entre naturalidade e consistência, evitando variação excessiva na formulação das perguntas sem torná-las artificiais. O *prompt* sistêmico impõe restrições de estilo (uma única frase, sem prefixos/listas) e orienta o uso de dicas de formato (por exemplo, "(AAAA-MM-DD)" para datas) e opções explícitas (por exemplo, "CPF ou CNS"). Para construir essas pistas, o agente aplica heurísticas locais:

- *_format_hint(slot)*: injeta dicas sintéticas conforme o tipo (*date, email, phone, postalCode, CPF, CNS, latitude, longitude*);
- *_options_hint(slot)*: identifica conjuntos de valores a partir de *value_set_url* (por exemplo, gênero administrativo; canais de contato; CPF/CNS).

Além disso, o *QuestionAgent* consulta as últimas interações (até 6 turnos) para evitar repetição e ajustar o enunciado ao contexto imediato. Quando a API não está disponível (ausência de chave), o agente executa um *fallback* determinístico, gerando uma pergunta mínima válida a partir do rótulo do campo, adicionando, quando houver, dica de formato e/ou opções. Essa estratégia garante resiliência do fluxo mesmo sob indisponibilidade temporária do LLM.

No ciclo colaborativo, o *OrchestratorAgent* seleciona o *slot* pendente e invoca o *QuestionAgent* para produzir a pergunta; após a resposta do usuário, o *InteractionAgent* executa a extração/normalização. Em casos de respostas ambíguas ou incompletas, o *QuestionAgent* volta a atuar com perguntas de esclarecimento (*follow-ups*), até que os *slots* obrigatórios estejam adequadamente preenchidos. Esse desacoplamento entre lógica determinística de fluxo (orquestração) e geração de linguagem natural controlada por política (perguntas) melhora a manutenibilidade, reuso e qualidade pragmática do diálogo.

O trecho de código abaixo ilustra a estrutura do *QuestionAgent* (*prompt*, heurísticas e *fallback*).

```
    import os
from typing import Optional, List

from langchain_openai import ChatOpenAI
from langchain_core.prompts import ChatPromptTemplate
```

```
from langchain_core.output_parsers import StrOutputParser

from src.agents.agent_interface import AgentInterface
from src.models.interview import SlotSpec

QUESTION_SYSTEM = """
Você é um entrevistador cordial e objetivo. Faça perguntas curtas, naturais e
claras.
- Use o contexto recente para evitar repetir o que já foi dito.
- Se houver opções, ofereça-as no próprio enunciado (ex.: "CPF ou CNS?").
- Se houver dica de formato, inclua em parênteses (curta), ex.: "(AAAA-MM-DD)".
- Escreva APENAS a pergunta ao usuário (uma frase). Sem prefixos, listas ou
explicações.
"""

def _format_hint(slot: SlotSpec) -> str:
    t = (slot.type or "").lower()
    sid = (slot.slot_id or "").lower()

    if t == "date":
        return "(AAAA-MM-DD)"
    if t == "email":
        return "(ex.: nome@dominio.com)"
    if t == "phone":
        return "(ex.: +55 62 91234-5678)"
    if t == "postalcode":
        return "(ex.: 75060-040 ou 75060040)"
    if t == "cpf":
        return "(11 dígitos, só números)"
    if t == "cns":
        return "(15 dígitos, só números)"
    if "latitude" in sid:
        return "(número decimal entre -90 e 90)"
    if "longitude" in sid:
        return "(número decimal entre -180 e 180)"
    return ""

def _options_hint(slot: SlotSpec) -> str:
    vs = getattr(slot, "value_set_url", None) or ""
```

```
    vs = vs.lower()

    # heurísticas simples
    if "sid/cpf" in vs or "sid/cns" in vs:
        return "CPF ou CNS"
    if "administrative-gender" in vs or "valueset-administrative-gender" in vs:
        return "Masculino, Feminino ou Outro"
    if "valueset-contact-point-system" in vs or "contact-point-system" in vs:
        return "telefone ou e-mail"
    return ""

class QuestionAgent(AgentInterface):
    def __init__(self, model: str = "gpt-4o-mini"):
        api_key = os.getenv("OPENAI_API_KEY")
        self.llm = ChatOpenAI(model=model, api_key=api_key, temperature=0.2) if
        api_key else None
        self.parser = StrOutputParser()

    async def render(self, slot: SlotSpec, previous_history: Optional[List[dict]]
    = None) -> str:
        # fallback determinístico (sem LLM)
        if not self.llm:
            label = slot.label or slot.slot_id
            hint = _format_hint(slot)
            opts = _options_hint(slot)
            if opts and hint:
                return f"{label} {hint} ({opts})"
            if opts:
                return f"{label} ({opts})"
            if hint:
                return f"{label} {hint}"
            return label

        hist = previous_history or []
        last_turns = "\n".join(
            f"{m.get('from')}: {m.get('text')}" for m in hist[-6:]
            if isinstance(m, dict) and m.get("text")
        )
```

```
        label = slot.label or slot.slot_id
        hint = _format_hint(slot)
        opts = _options_hint(slot)

        extra = []
        if opts:
            extra.append(f"Opções: {opts}")
        if hint:
            extra.append(f"Dica de formato: {hint}")

        human = f"""
Campo:
- id: {slot.slot_id}
- rótulo: {label}
- tipo: {slot.type}
- obrigatório: {slot.required}
{('- ' + ' / '.join(extra)) if extra else ''}

Histórico recente (use para evitar soar repetitivo):
{last_turns or '-'}

Gere UMA pergunta natural e direta para coletar esse campo.
Inclua as opções e/ou a dica de formato quando cabível, em no máximo 20 palavras.
A resposta deve ser APENAS a pergunta (uma única frase).
""".strip()

        prompt = ChatPromptTemplate.from_messages([("system", QUESTION_SYSTEM),
        ("human", human)])
        chain = prompt | self.llm | self.parser
        return await chain.ainvoke({})

    async def execute(self, *args, **kwargs) -> dict:
        return {}
```

4.3 Interface Conversacional (Streamlit e WhatsApp)

A validação prática da arquitetura de coleta conversacional foi demonstrada por sessões reais de coleta, suportadas por duas frentes de interação: (i) uma interface *web* em *Streamlit*, utilizada como ambiente de teste, validação e monitoramento, e (ii) um canal de mensageria via *WhatsApp* (integração *EvolutionAPI*), responsável pela ingestão reativa

de mensagens em tempo real. A demonstração ponta a ponta abrangeu desde a captura do texto livre até a persistência do recurso FHIR correspondente, permitindo observar o comportamento do protótipo sob condições de uso realistas.

4.3.1 Interface *Web* (*Streamlit*)

A interface em *Streamlit* disponibilizou *dashboards* operacionais para cadastro / visualização de artefatos FHIR (Figura 9), inspeção do histórico conversacional (Figura 10) e acionamento de fluxos de coleta (Figura 11). Esse ambiente foi usado para: (a) validar dados estruturados antes do envio ao servidor FHIR, (b) acompanhar estados de sessão (campos preenchidos/pendentes) e (c) verificar em tempo real a normalização produzida pelo pipeline de agentes.

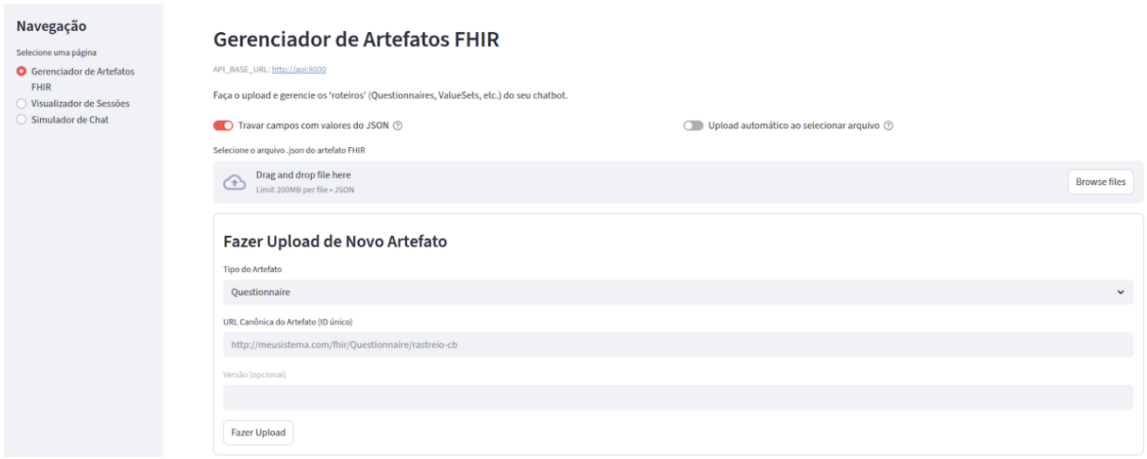

Figura 9 – Interface de cadastro de artefatos FHIR (*Streamlit*).

Figura 10 – Interface para visualizar o histórico de conversas no (*Streamlit*).

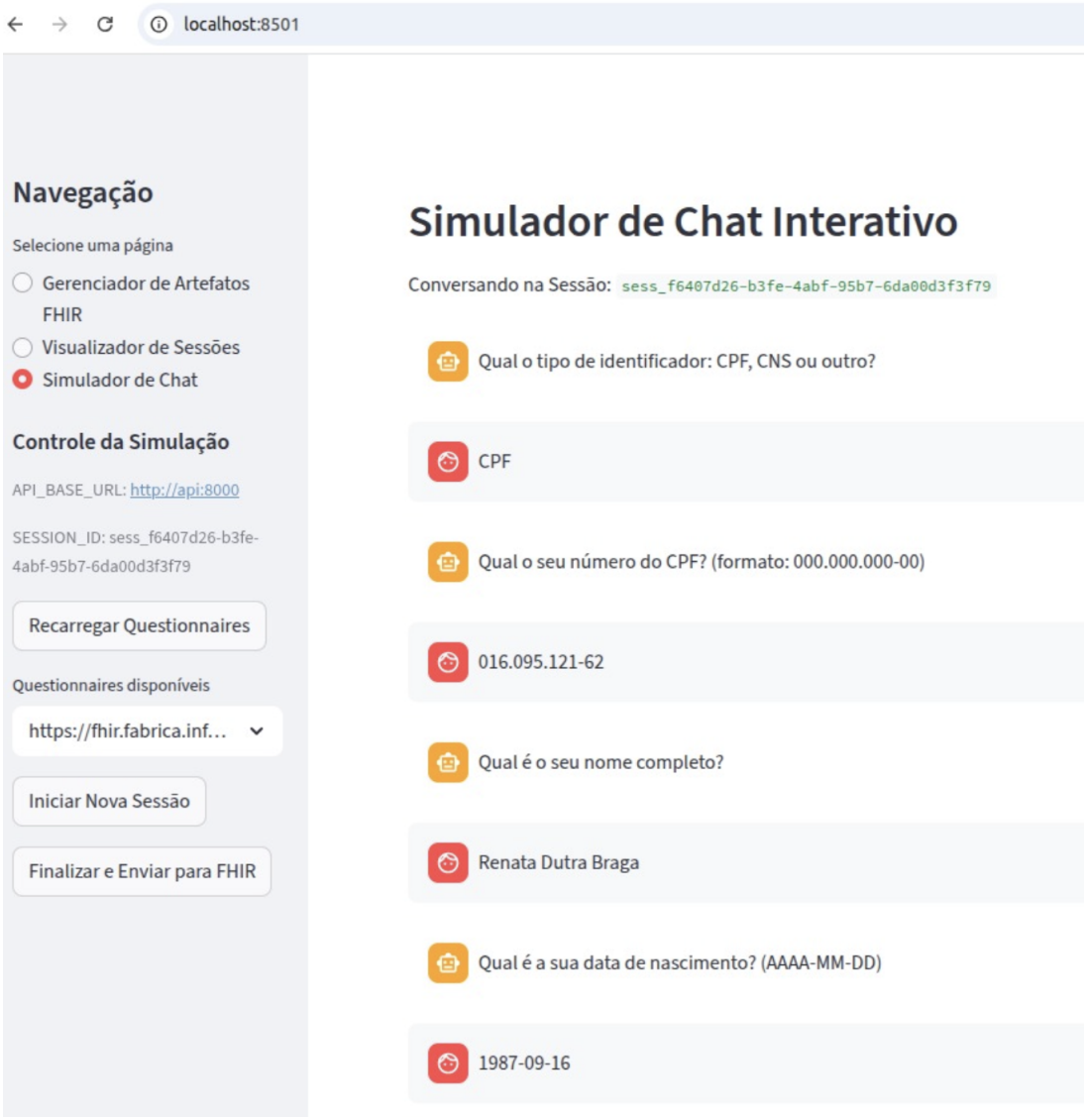

Figura 11 – Interface de conversação via web (*Streamlit Chat*).

4.3.2 Integração com *WhatsApp* (*EvolutionAPI*): ingestão reativa por *webhooks*

O canal *WhatsApp* foi integrado por meio de *webhooks* fornecidos pela *EvolutionAPI*, operacionalizando um modelo orientado a eventos (Figura 12). O *webhook* foi implementado na *FastAPI*, que realizou a validação do *token*, o roteamento de eventos *messages.upsert* para o *ConversationService* e a atualização do estado de sessão no MongoDB. Os *Endpoints* *POST /conversation/start-whatsapp* também foram expostos pela *FastAPI* para acionar coletas programáticas. O *endpoint* principal, *POST /webhook/evolution*, atuou como porta de entrada para as mensagens recebidas:

- Autenticação por *token* (via *query parameter*): somente requisições válidas foram processadas;

- Filtragem de eventos: a solução processou especificamente *messages.upsert* (novas mensagens) e ignorou ruídos operacionais (status de entrega/presença);
- Foco em conversas individuais: mensagens de grupo foram deliberadamente descartadas, mantendo o escopo *one-to-one*.

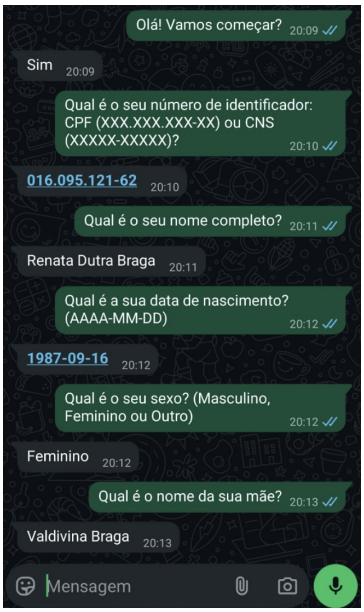

Figura 12 – Interface de conversação via. (*WhatsApp*).

Essas evidências mostram a coerência do diálogo nos dois canais (*web* e *WhatsApp*), com a mesma política de orquestração e geração de perguntas.

Para manter o contexto entre múltiplas interações, foi adotado um mapa em memória (*SESSION_BY_NUMBER*), que associa cada número de *WhatsApp* a um *session_id* único (Figura 13). Quando não havia sessão ativa, a solução criava automaticamente uma nova via *POST /conversation/sessions* e inicializava o fluxo com um comando padrão (*/start*) atrelado ao questionário FHIR pertinente.

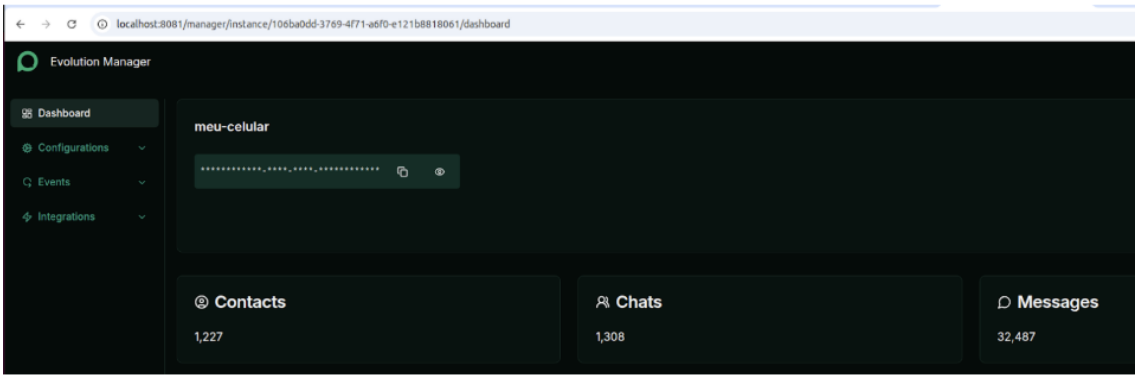

Figura 13 – Número de *WhatsApp* torna-se o *session_id*.

Além do fluxo puramente reativo, a solução disponibilizou o endpoint *POST /conversation/start-whatsapp* para inicialização programática de coletas. Esse ponto de entrada aceita parâmetros como número de destino e URL do questionário FHIR, viabilizando estratégias ativas de engajamento.

4.4 Persistência FHIR e comprovação de interoperabilidade

Ao término da coleta, os dados foram mapeados automaticamente para o recurso FHIR Patient (e demais artefatos, quando aplicável) (Figura 14) e submetidos ao HAPI FHIR Server (Figura 15). O envio bem-sucedido, após as validações de cardinalidade/semântica do perfil, comprovou a interoperabilidade do pipeline com repositórios clínicos padronizados.

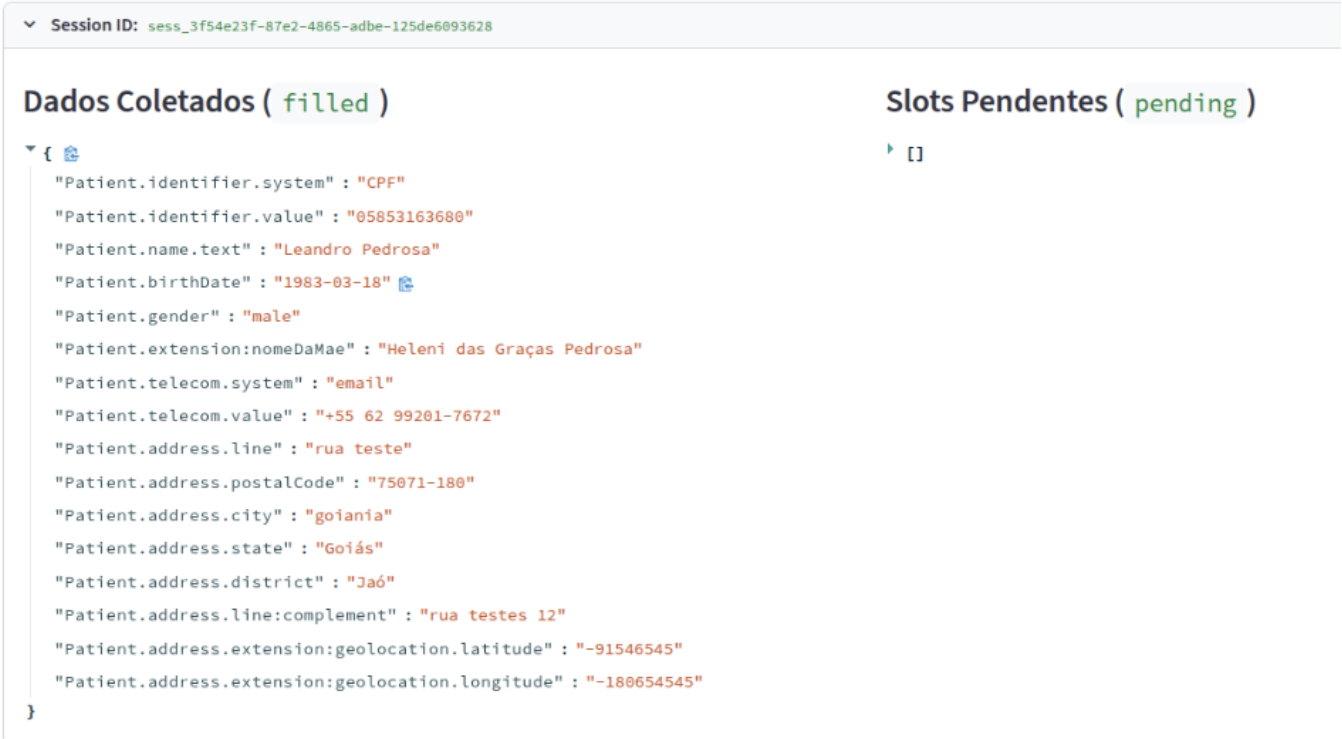

Figura 14 – Dados coletados da conversa.

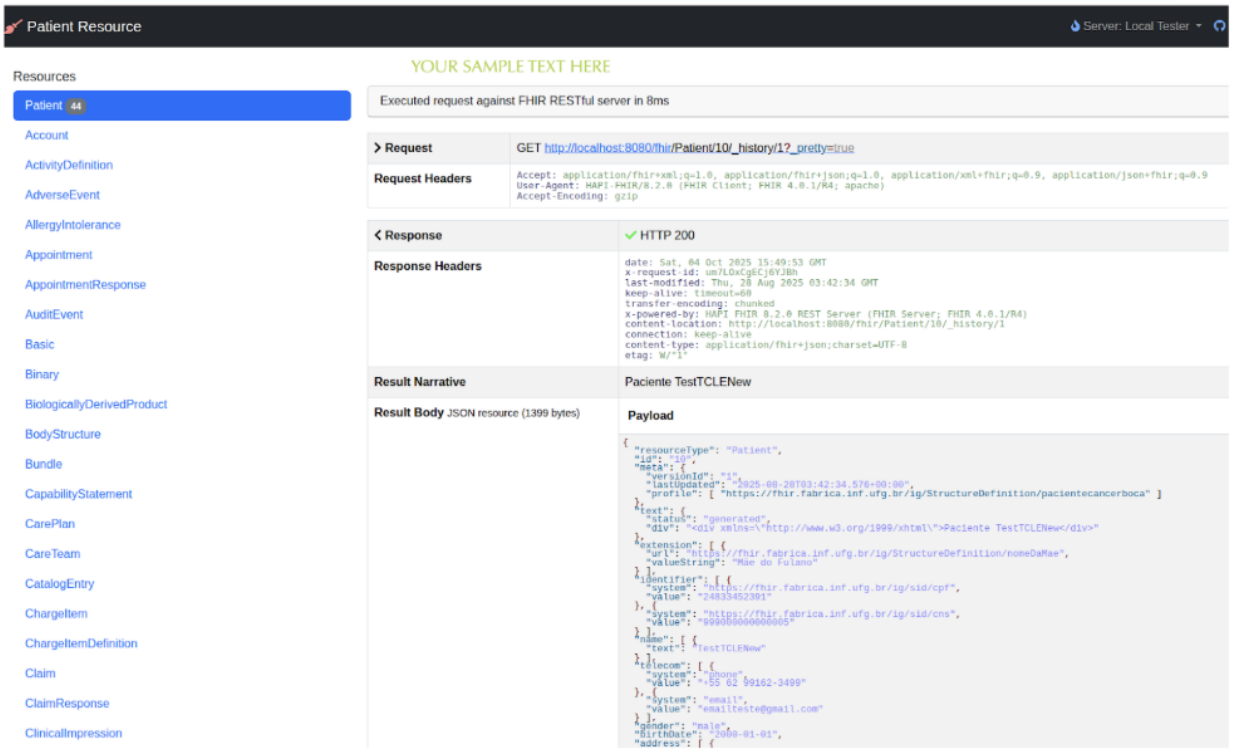

Figura 15 – Recurso FHIR Patient persistido no servidor HAPI FHIR. Os dados destacados foram extraídos, normalizados e estruturados automaticamente pelo *pipeline* de IA, validando o processo de ponta a ponta.

As figuras destacam os campos extraídos, normalizados e estruturados pelo pipeline, fechando o circuito entre mensageria, agentes, validação e persistência clínica. Assim, a interface conversacional comprovou a viabilidade prática do protótipo no cenário-alvo: condução de entrevistas clínicas em linguagem natural, com memória de sessão, normalização e persistência FHIR. A união entre orquestração determinística, geração e compreensão de linguagem natural e validação estruturada consolidou um fluxo simultaneamente natural para o usuário e confiável do ponto de vista informacional, atendendo ao objetivo metodológico desta etapa do estudo.

4.5 Avaliação dos Resultados

A etapa de avaliação teve como foco verificar a eficácia funcional e técnica do *pipeline* conversacional proposto, assegurando que todos os componentes. desde a coleta em linguagem natural até a persistência dos recursos FHIR, operassem de forma integrada, segura e consistente.

O processo de validação foi conduzido por meio de execuções controladas do protótipo, abrangendo tanto a interface *web* (*Streamlit*) quanto o canal de mensageria (*WhatsApp* via *EvolutionAPI*). Em ambos os contextos, o comportamento da solução foi

monitorado desde o recebimento das mensagens até a gravação dos dados estruturados no servidor HAPI FHIR. Essa verificação permitiu confirmar a coerência entre os fluxos reativos (*webhook*) e os fluxos programáticos (inicialização via *endpoint*), demonstrando a confiabilidade operacional do *pipeline* frente a múltiplos pontos de entrada e modos de interação.

Durante os testes, foram observados os seguintes resultados:

- Captura e autenticação de mensagens: as mensagens recebidas pelo *webhook* do *EvolutionAPI* foram corretamente validadas por *token* de segurança e filtradas pelo tipo de evento (*messages.upsert*), descartando notificações não relevantes (status de entrega, presença, entre outros);
- Gestão inteligente de sessões: o mecanismo *SESSION_BY_NUMBER* garantiu a manutenção do contexto conversacional, associando cada número de *WhatsApp* a um *session_id* único e permitindo a retomada de diálogos interrompidos, em conformidade com o estado persistido no MongoDB (*SessionState*);
- Orquestração dos agentes: o *ConversationService* integrou o *OrchestratorAgent* e o *InteractionAgent* de forma coerente, assegurando a sequência lógica das perguntas e o tratamento semântico das respostas;
- Conversão e validação dos dados: as respostas dos usuários foram transformadas em objetos JSON estruturados, validados quanto à consistência semântica e convertidos automaticamente em recursos FHIR (*Patient*, *QuestionnaireResponse*);
- Persistência interoperável: os recursos foram submetidos ao servidor HAPI FHIR e armazenados com sucesso, comprovando interoperabilidade sintática (formato JSON conforme o padrão HL7® FHIR R4) e semântica (valores normalizados conforme o modelo de informação definido);
- Resiliência operacional; a solução demonstrou tolerância a erros de formatação e conectividade, incluindo *fallbacks* e revalidação automática de entradas inconsistentes, conforme previsto no código do *evolution_webhook.py*.

4.6 Trabalhos Futuros

A continuidade deste estudo abre diversas perspectivas de pesquisa e desenvolvimento tecnológico, as quais concentram-se em tornar a solução aplicável a múltiplas patologias e validado em contextos reais.

Uma primeira direção consiste na expansão da arquitetura proposta para o rastreamento e monitoramento de outras doenças crônicas não transmissíveis, como por exemplo, diabetes *mellitus*, hipertensão arterial sistêmica, doença pulmonar obstrutiva crônica,

doenças cardiovasculares, insuficiência renal crônica e até câncer de próstata ou de colo uterino. Nesses cenários, o mesmo núcleo conversacional poderá ser adaptado para coletar dados clínicos e comportamentais de cada condição, mediante a criação de novos modelos de informação e perfis FHIR correspondentes, preservando a interoperabilidade semântica entre diferentes sistemas de informação em saúde e o potencial de integração com a RNDS.

Outra linha de evolução possível contempla a validação do *chatbot* em ambiente real de utilização, envolvendo profissionais e usuários do SUS, de modo a avaliar os aspectos de usabilidade, aceitabilidade e desempenho clínico. Essa etapa permitirá observar a interação da solução com dados reais, mensurar indicadores de qualidade da coleta e identificar ajustes necessários para garantir a confiabilidade, segurança e aderência à LGPD.

Além disso, recomenda-se investigar o uso de modelos de linguagem abertos e especializados em saúde, capazes de operar localmente ou em infraestruturas seguras, reduzindo custos operacionais e ampliando a transparência do processo inferencial (Magnini; Aguzzi; Montagna, 2025). Complementarmente, pode-se incorporar mecanismos de explicabilidade e revisão humana assistida, voltados à mitigação de vieses e à validação de informações clínicas sensíveis (Abbas; Jeong; Lee, 2025).

Por fim, vislumbra-se a integração do *chatbot* com painéis de monitoramento inteligente e sistemas de apoio à decisão, permitindo o acompanhamento longitudinal de pacientes e a geração de alertas precoces baseados em dados coletados em tempo real. Essas extensões consolidam o potencial da solução como componente modular de um ecossistema interoperável e escalável para a saúde digital orientada por inteligência artificial (Helminski *et al.*, 2022; Barreda *et al.*, 2025; Griffin *et al.*, 2021; Altom *et al.*, 2025).

5 CONCLUSÕES

O presente estudo desenvolveu uma arquitetura multiagente baseada em modelos de linguagem de grande escala (LLMs), capaz de conduzir a coleta estruturada de informações clínicas e demográficas em conformidade com o padrão HL7® FHIR®. O protótipo desenvolvido demonstrou a viabilidade técnica da abordagem proposta, comprovando que é possível transformar interações em linguagem natural, realizadas via interface *web* ou mensageria (*WhatsApp*), em recursos FHIR computáveis, interoperáveis e semanticamente consistentes.

A arquitetura conteinerizada, composta por módulos independentes (*EvolutionAPI*, API central, MongoDB e HAPI FHIR), mostrou-se adequada para o processamento de dados heterogêneos e para a integração em ecossistemas de saúde digital. A utilização de uma camada de orquestração determinística (*OrchestratorAgent*), juntamente com os agentes especializados em geração e interpretação de linguagem (*QuestionAgent* e *InteractionAgent*), permitiu estruturar um fluxo conversacional previsível, rastreável e capaz de garantir completude mínima dos dados.

Os testes empíricos realizados em ambiente controlado validaram o funcionamento ponta a ponta do *pipeline* conversacional, abrangendo desde a ingestão de mensagens e manutenção de contexto de sessão até a conversão e persistência dos dados no servidor HAPI FHIR. As respostas coletadas foram corretamente estruturadas em formato JSON, validadas quanto a formato e semântica, e transformadas em recursos *Patient* compatíveis com o modelo FHIR, assegurando interoperabilidade e auditabilidade.

Do ponto de vista metodológico, o estudo demonstrou que a integração entre inteligência probabilística, via LLM, e regras determinísticas de validação é uma estratégia eficaz para garantir a qualidade e a confiabilidade dos dados em sistemas conversacionais voltados à saúde. Essa solução híbrida superou limitações de métodos convencionais de extração e transformação de dados textuais, ao mesmo tempo em que evitou inconsistências típicas de modelos generativos operando sem restrições.

Embora não tenham sido aplicadas métricas quantitativas de desempenho (como tempo de resposta ou precisão de extração), a avaliação funcional evidenciou a coerência lógica do fluxo, validando o protótipo como uma prova de conceito sólida para futuras implementações em larga escala.

Como principais contribuições, destacam-se:

- A concepção de uma arquitetura conversacional interoperável compatível com o padrão HL7® FHIR®;

- O desenvolvimento de um *pipeline* de extração e normalização de dados clínicos via LLM, com geração automática de recursos FHIR;
- A demonstração de um modelo de orquestração multiagente aplicável à coleta distribuída de dados em ambientes de saúde digital;
- A integração bem-sucedida de canais de interação distintos (*web* e *WhatsApp*), preservando o contexto e a consistência dos dados coletados.

Entre as limitações, ressalta-se que o protótipo concentrou-se em dados cadastrais básicos, não abrangendo o processamento de informações clínicas complexas ou terminologias clínicas especializadas. Além disso, o uso de modelos proprietários (*OpenAI GPT-4o-mini*) implica restrições de custos e controle sobre o treinamento. Trabalhos futuros devem explorar alternativas *open-source* otimizadas para o domínio da saúde, avaliar métricas de desempenho e incorporar mecanismos de mecanismos de revisão humana assistida para revisão de dados de baixa confiança, bem como camadas adicionais de segurança e conformidade com a LGPD.

Em síntese, o estudo comprovou que a união de agentes inteligentes, interoperabilidade semântica e engenharia de dados estruturada pode formar a base de sistemas conversacionais confiáveis e escaláveis, aptos a apoiar processos de coleta, qualificação e integração de informações em saúde digital. Essa contribuição representa um avanço em direção a soluções mais acessíveis, automatizadas e interoperáveis no contexto da Estratégia de Saúde Digital para o Brasil (2020–2028).

REFERÊNCIAS

ABBAS, Q.; JEONG, W.; LEE, S. W. Explainable ai in clinical decision support systems: A meta-analysis of methods, applications, and usability challenges. **Healthcare (Basel, Switzerland)**, MDPI, v. 13, n. 17, p. 2154, 2025. Disponível em: https://doi.org/10.3390/healthcare13172154. Acesso em: 05 out. 2025.

ALBERT, A.; TIZZARD, E. Large language models for improving cancer diagnosis and management in primary health care settings. **Journal of Medicine, Surgery, and Public Health**, Elsevier, v. 4, p. 100157, 2024. ISSN 2949-916X. Disponível em: https://www.sciencedirect.com/science/article/pii/S2949916X24001105. Acesso em: 12 set. 2025.

ALTOM, D. S. *et al.* Artificial intelligence-based chatbots in chronic disease management: A systematic review of applications and challenges. **Cureus**, Cureus, Inc., v. 17, n. 3, p. e81001, 2025. Disponível em: https://doi.org/10.7759/cureus.81001. Acesso em: 12 set. 2025.

AYDIN, S. *et al.* Large language models in patient education: a scoping review of applications in medicine. **Frontiers in Medicine**, Frontiers Media, v. 11, p. 1477898, 2024. Disponível em: https://doi.org/10.3389/fmed.2024.1477898. Acesso em: 12 set. 2025.

BARREDA, M. *et al.* Transforming healthcare with chatbots: Uses and applications – a scoping review. **Digital Health**, SAGE Publications, v. 11, p. 20552076251319174, 2025. Disponível em: https://doi.org/10.1177/20552076251319174. Acesso em: 05 out. 2025.

Brasil. **Estratégia de Saúde Digital para o Brasil 2020–2028 [recurso eletrônico]**. Brasília: Ministério da Saúde. Secretaria-Executiva. Departamento de Informática do SUS, 2020. 128 p. Il. ISBN 978-85-334-2841-6. Disponível em: https://bvsms.saude.gov.br/bvs/publicacoes/estrategia_saude_digital_Brasil.pdf. Acesso em: 12 set. 2025.

BRIGANTI, G. How chatgpt works: a mini review. **European Archives of Oto-Rhino-Laryngology**, Springer, v. 281, p. 1565–1569, 2024. Disponível em: https://doi.org/10.1007/s00405-023-08337-7. Acesso em: 12 set. 2025.

BROWN, T. B. *et al.* **Language Models are Few-Shot Learners**. 2020. Disponível em: https://arxiv.org/abs/2005.14165.

BUBECK, S. *et al.* Sparks of artificial general intelligence: Early experiments with GPT-4. **arXiv preprint arXiv:2303.12712**, 2023. Disponível em: https://doi.org/10.48550/arXiv.2303.12712. Acesso em: 12 set. 2025.

CHAMOLI, A.; AL. et. Overview of oral cavity squamous cell carcinoma: Risk factors, mechanisms, and diagnostics. **Oral Oncology**, Elsevier, v. 121, p. 105451, 2021. Disponível em: https://doi.org/10.1016/j.oraloncology.2021.105451. Acesso em: 05 out. 2025.

CLAMAN, D.; SEZGIN, E. Artificial intelligence in dental education: Opportunities and challenges of large language models and multimodal foundation models. **JMIR**

Medical Education, JMIR Publications, v. 10, p. e52346, 2024. Disponível em: https://doi.org/10.2196/52346. Acesso em: 12 set. 2025.

GIANNAKOPOULOS, K. *et al.* Evaluation of the performance of generative ai large language models chatgpt, google bard, and microsoft bing chat in supporting evidence-based dentistry: Comparative mixed methods study. **Journal of Medical Internet Research**, JMIR Publications, v. 25, p. e51580, 2023. Disponível em: https://doi.org/10.2196/51580. Acesso em: 12 set. 2025.

GRIFFIN, A. C. *et al.* Conversational agents for chronic disease self-management: A systematic review. *In*: AMIA SYMPOSIUM. **AMIA Annual Symposium Proceedings**. [*S.l.: s.n.*]: American Medical Informatics Association, 2021. p. 504–513. Proceedings of the AMIA 2020 Annual Symposium. Acesso em: 05 out. 2025.

HELMINSKI, D. *et al.* Dashboards in health care settings: Protocol for a scoping review. **JMIR Research Protocols**, JMIR Publications, v. 11, n. 3, p. e34894, 2022. Disponível em: https://doi.org/10.2196/34894. Acesso em: 05 out. 2025.

HL7 International. **FHIR Release 4 – Technical Specification**. 2023. Online. Acesso em: 12 set. 2025. Disponível em: https://www.hl7.org/fhir/.

HUSSAIN, M. A.; LANGER, S. G.; KOHLI, M. Learning hl7 fhir using the hapi fhir server and its use in medical imaging with the siim dataset. **Journal of Digital Imaging**, Springer, v. 31, n. 3, p. 334–340, 2018. Disponível em: https://doi.org/10.1007/s10278-018-0090-y. Acesso em: 12 set. 2025.

INCA. **Síntese de Resultados e Comentários: Câncer da Cavidade Oral**. 2023. Página eletrônica. Instituto Nacional de Câncer. Acesso em: 12 set. 2025. Disponível em: https://www.gov.br/inca/pt-br/assuntos/cancer/numeros/estimativa/sintese-de-resultados-e-comentarios.

JI, Z. *et al.* Survey of hallucination in natural language generation. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 55, n. 12, mar. 2023. ISSN 0360-0300. Disponível em: https://doi.org/10.1145/3571730.

KHOSHROUDI, S. H. N.; SAFAEI, A. A.; SOLEIMANJAHI, H. A nosql document based ecrf system for study of vaccines with variable adverse events: case study on covid-19 vaccines. **Scientific Reports**, Nature Publishing Group, v. 15, p. 20453, 2025. Disponível em: https://doi.org/10.1038/s41598-025-05746-y. Acesso em: 05 out. 2025.

KURNIAWAN, M. H. *et al.* A systematic review of artificial intelligence-powered (ai-powered) chatbot intervention for managing chronic illness. **Annals of Medicine**, Taylor & Francis, v. 56, n. 1, p. 2302980, 2024. Disponível em: https://doi.org/10.1080/07853890.2024.2302980. Acesso em: 12 set. 2025.

LARANJO, L. *et al.* Conversational agents in healthcare: a systematic review. **Journal of the American Medical Informatics Association (JAMIA)**, Oxford University Press, v. 25, n. 9, p. 1248–1258, 2018. Disponível em: https://doi.org/10.1093/jamia/ocy072. Acesso em: 12 set. 2025.

LI, X. *et al.* A survey on llm-based multi-agent systems: workflow, infrastructure, and challenges. **Vicinagearth**, Springer Nature, v. 1, p. 9, 2024. Disponível em: https://doi.org/10.1007/s44336-024-00009-2. Acesso em: 05 out. 2025.

LI, Y. *et al.* Fhir-gpt enhances health interoperability with large language models. **NEJM AI**, Massachusetts Medical Society, v. 1, n. 8, p. 10.1056/aics2300301, 2024. Disponível em: https://doi.org/10.1056/aics2300301. Acesso em: 05 out. 2025.

MAGNINI, M.; AGUZZI, G.; MONTAGNA, S. Open-source small language models for personal medical assistant chatbots. **Intelligence-Based Medicine**, Elsevier, v. 11, p. 100197, 2025. Disponível em: https://www.sciencedirect.com/science/article/pii/S2666521224000644. Acesso em: 05 out. 2025.

MANDEL, J. C. *et al.* Smart on fhir: a standards-based, interoperable apps platform for electronic health records. **Journal of the American Medical Informatics Association (JAMIA)**, Oxford University Press, v. 23, n. 5, p. 899–908, 2016. Disponível em: https://doi.org/10.1093/jamia/ocv189. Acesso em: 12 set. 2025.

MESKÓ, B.; GÖRÖG, M. A short guide for medical professionals in the era of artificial intelligence. **npj Digital Medicine**, Nature Publishing Group, v. 3, p. 126, 2020. Disponível em: https://doi.org/10.1038/s41746-020-00333-z. Acesso em: 12 set. 2025.

OBERMEYER, Z. *et al.* Dissecting racial bias in an algorithm used to manage the health of populations. **Science**, American Association for the Advancement of Science, v. 366, n. 6464, p. 447–453, 2019. Disponível em: https://doi.org/10.1126/science.aax2342. Acesso em: 12 set. 2025.

O'CONNOR, B. D. *et al.* The dockstore: enabling modular, community-focused sharing of docker-based genomics tools and workflows. **F1000Research**, F1000 Research Ltd, v. 6, p. 52, 2017. Disponível em: https://doi.org/10.12688/f1000research.10137.1. Acesso em: 05 out. 2025.

OSHIN, M.; CAMPOS, N. **Learning LangChain: Building AI and LLM Applications with LangChain and LangGraph**. 1. ed. Sebastopol, CA: O'Reilly Media, 2025. 294 p. ISBN 978-1098167288. Acesso em: 12 set. 2025.

PALUMBO, R.; NICOLA, C.; ADINOLFI, P. Addressing health literacy in the digital domain: insights from a literature review. **Kybernetes**, Emerald Publishing, v. 51, n. 13, p. 82–97, 2022. Disponível em: https://doi.org/10.1108/K-07-2021-0547. Acesso em: 12 set. 2025.

PEDROSA, L. *et al.* Desenvolvimento de aplicativo para monitorar risco de câncer de boca. **Journal of Health Informatics**, Sbis, v. 16, n. Especial, 2024. Disponível em: https://doi.org/10.59681/2175-4411.v16.iEspecial.2024.1269. Acesso em: 12 set. 2025.

SANTOS, M. d. O. *et al.* Estimativa de incidência de câncer no brasil, 2023–2025. **Revista Brasileira de Cancerologia**, Instituto Nacional de Câncer (INCA), v. 69, n. 1, p. e–213700, fev. 2023. Publicado em: 06 fev. 2023. [Citado em: 12 set. 2025.]. Disponível em: https://rbc.inca.gov.br/index.php/revista/article/view/3700.

SATO, A. *et al.* Preliminary screening for hereditary breast and ovarian cancer using a chatbot augmented intelligence genetic counselor: Development and feasibility study. **JMIR Formative Research**, JMIR Publications, v. 5, n. 2, p. e25184, 2021. Disponível em: https://doi.org/10.2196/25184. Acesso em: 12 set. 2025.

SCHMIEDMAYER, P. *et al.* **LLM on FHIR – Demystifying Health Records**. 2024. Disponível em: https://arxiv.org/abs/2402.01711.

SEN, P. S.; MUKHERJEE, N. An ontology-based approach to designing a nosql database for semi-structured and unstructured health data. **Cluster Computing**, Springer Nature, p. 1–18, 2023. Advance online publication. Disponível em: https://doi.org/10.1007/s10586-023-03995-y. Acesso em: 05 out. 2025.

SHAH, S. M.; KHAN, R. A. Secondary use of electronic health record: Opportunities and challenges. **IEEE Access**, v. 8, p. 136947–136965, 2020.

SINGHAL, K. *et al.* Toward expert-level medical question answering with large language models. **Nature Medicine**, Nature Publishing Group, v. 31, p. 943–950, 2025. Disponível em: https://doi.org/10.1038/s41591-024-03423-7. Acesso em: 12 set. 2025.

SPEIGHT, P. M. *et al.* Screening for oral cancer–a perspective from the global oral cancer forum. **Oral Surgery, Oral Medicine, Oral Pathology and Oral Radiology**, Elsevier, v. 123, n. 6, p. 680–687, jun. 2017. Disponível em: https://doi.org/10.1016/j.oooo.2016.08.021. Acesso em: 12 set. 2025.

THIRUNAVUKARASU, A. J. *et al.* Large language models in medicine. **Nature Medicine**, Nature Publishing Group, v. 29, n. 8, p. 1930–1940, 2023. Disponível em: https://doi.org/10.1038/s41591-023-02448-8. Acesso em: 12 set. 2025.

TOPOL, E. **Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again**. 1. ed. New York: Basic Books, 2019. 400 p. ISBN 978-1541644632. Acesso em: 12 set. 2025.

VASWANI, A. *et al.* Attention is all you need. **CoRR**, abs/1706.03762, 2017. Disponível em: http://arxiv.org/abs/1706.03762.

WHO. **Global Strategy on Digital Health 2020–2025**. Genebra: World Health Organization, 2021. Documento PDF. World Health Organization. © World Health Organization 2021. Acesso em: 12 set. 2025. Disponível em: https://www.who.int/docs/default-source/documents/gs4dhdaa2a9f352b0445bafbc79ca799dce4d.pdf.

WORKMAN, A. D. *et al.* Utility of a langchain and openai gpt-powered chatbot based on the international consensus statement on allergy and rhinology: Rhinosinusitis. **International Forum of Allergy & Rhinology**, Wiley, v. 14, p. 1101–1109, 2024. Disponível em: https://doi.org/10.1002/alr.23310. Acesso em: 12 set. 2025.

YANG, R. *et al.* Large language models in health care: development, applications, and challenges. **Health Care Science**, Wiley, v. 2, p. 255–263, 2023. Disponível em: https://doi.org/10.1002/hcs2.61. Acesso em: 12 set. 2025.

APÊNDICES

APÊNDICE A – MODELO DE INFORMAÇÃO – PACIENTE DO GRUPO DE RISCO PARA CÂNCER DE BOCA

Tabela 3 – Modelo de Informação (MI) - Paciente

Nível	**Card.**	**Elemento de Dados**	**Tipo de Dados**	**Descrição / observação / regras de negócio**	**Recursos semânticos (CodeSystem / ValueSet / Terminologias. . .)**	**Recursos / perfis FHIR**
1	[1..1]	Informações sociodemográficas	Seção	Coleta de dados básicos de identificação e demografia do paciente.		
2	[1..2]	Identificação do paciente	Subseção	RN01: O paciente deve ser identificado pelo CPF e/ou CNS.		
3	[0..1]	Tipo de identificador do paciente	URI	Define o namespace para o valor do identificador do paciente.	CNS: <url> https://fhir.fabrica.inf.ufg.br/cb/sid/cns CPF: <url> https://fhir.fabrica.inf.ufg.br/cb/sid/cpf	Patient.identifier.system
3	[0..1]	Identificador Nacional do paciente	Caracteres numéricos	Número do identificador único do paciente. RN02: Os Identificadores Nacionais do paciente (CPF ou CNS) devem possuir um número válido.		Patient.identifier.value
2	[1..2]	Região de saúde vinculada ao paciente	Subseção	Coleta os dados pessoais do paciente.		
3	[0..1]	Microrregiao de saúde	URI	Define o namespace para a microrregião de saúde que o paciente reside.		Location.physicalType. coding.system
3	[0..1]	Unidade de Saúde de Origem	Referência ao Estabelecimento de Saúde	Descreve o estabelecimento de saúde responsável pelo registro primário do paciente (UBS de referência da origem do paciente)		Patient.managingOrganization
2	[1..1]	Dados pessoais	Subseção	Coleta os dados pessoais do paciente.		
3	[1..1]	Nome do paciente	Sequência de caracteres alfanuméricos	Nome completo do paciente cadastrado.		Patient.name.text
3	[1..1]	Data de nascimento	Data	Data de nascimento do paciente, no formato ISO 8601 (YYYY-MM-DD).		Patient.birthDate
3	[1..1]	Sexo	Texto codificado	Sexo de um paciente ao nascer, sendo: Feminino Masculino Outro Personalizar a partir de CodeSystem	https://hl7.org/fhir/R4/valueset-administrative-gender.html	Patient.gender
3	[1..1]	Nome da mãe	Sequência de caracteres alfanuméricos	Nome completo da mãe do paciente, conforme registrado no sistema.		Patient.extension
2	[1..N]	Meios de contato	Subseção	Fornece as formas de contato com o paciente (telefone celular, telefone do responsável e e-mail).		
3	[1..1]	Tipo de meio de contato	Texto codificado	phone: número de telefone do paciente email: endereço de correio eletrônico (e-mail) do paciente	ValueSet: https://hl7.org/fhir/R4/valueset-contact-point-system.html	Patient.telecom.system

Nível	Card.	Elemento de Dados	Tipo de Dados	Descrição / observação / regras de negócio	Recursos semânticos (CodeSystem / ValueSet / Terminologias. . .)	Recursos / perfis FHIR
3	[1..1]	Meio de contato	Sequência de caracteres alfanuméricos	Número de telefone ou e-mail do paciente. RN03: Validar o formato do número de telefone quando o tipo de meio de contato for igual a "phone". O número de telefone deve seguir o formato internacional, que inclui o código do país (+CC), seguido pelo código de área (DDD) e pelo número de telefone no formato +CC DDD XXXXX-XXXX ou +CC DDD XXXX-XXXX. Exemplos válidos seriam +55 62 91234-5678 ou +55 62 3011-5678. O número deve conter apenas dígitos e espaços permitidos, sem caracteres especiais, como letras ou símbolos RN04: Validar o formato do e-mail quando o tipo de meio de contato for igual a "email". O endereço de e-mail deve seguir a estrutura padrão, que consiste no local-part seguido de "@"e o domínio, como por exemplo: exemplo@provedor.com. O local-part pode conter letras, números, pontos, sublinhados e hífens, enquanto o domínio deve conter ao menos um ponto, seguido de uma extensão válida, como .com, .org ou .br. O e-mail não deve conter espaços ou caracteres especiais inválidos.	http://hl7.org/fhir/contact-point-system	Patient.telecom.value
2	[1..1]	Endereço	Subseção	Endereço completo onde o paciente reside ou pode ser localizado. Inclui CEP, Estado, Cidade, Complemento e Bairro.		
3	[1..1]	Endereço completo	Sequência de caracteres alfanuméricos	Dados do(s) endereço(s) onde o paciente pode ser localizado.		Patient.address.line
3	[1..1]	CEP	Sequência de caracteres alfanuméricos	Código postal do endereço do paciente.		Patient.address.postalCode
3	[0..1]	Complemento	Sequência de caracteres alfanuméricos	Informações adicionais sobre o endereço do paciente, como número, bloco, etc.		Patient.address.line
3	[0..1]	Bairro	Sequência de caracteres alfanuméricos	Bairro onde o paciente reside.		Patient.address.line
3	[0..1]	Cidade	Sequência de caracteres alfanuméricos	Cidade onde o paciente reside.		Patient.address.city
3	[0..1]	Estado	Sequência de caracteres alfanuméricos	Estado onde o paciente reside.		Patient.address.state
3	[0..1]	Latitude	Número decimal	Coordenada geográfica (latitude) do local de residência do paciente, no formato decimal.		Patient.address.extension. valueDecimal
3	[0..1]	Longitude	Número decimal	Coordenada geográfica (longitude) do local de residência do paciente, no formato decimal.		Patient.address.extension. valueDecimal
2	[1..1]	Fatores de Risco	Subseção	Fatores de Risco		
3	[1..N]	Fatores de Risco	Texto estruturado	Descreve os fatores de risco do paciente. Tabagista Etilista Homem maior que 50 anos Lesão bucal suspeita		RiskAsessment.subject.final RiskAsessment.Code

Tabela 3 – Modelo de Informação – Paciente do Grupo de Risco para Câncer de Boca