



UNIVERSIDADE FEDERAL DO CARIRI
CENTRO DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

CLICE BEZERRA BRITO ROMÃO

**PROTÓTIPO DE SISTEMA *WEB* PARA APOIO À DECISÃO CLÍNICA COM
CLASSIFICAÇÃO AUTOMATIZADA DE ANORMALIDADES EM EXAMES
GASTROINTESTINAIS**

JUAZEIRO DO NORTE - CE

2026

CLICE BEZERRA BRITO ROMÃO

PROTÓTIPO DE SISTEMA *WEB* PARA APOIO À DECISÃO CLÍNICA COM
CLASSIFICAÇÃO AUTOMATIZADA DE ANORMALIDADES EM EXAMES
GASTROINTESTINAIS

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Ciência da Computação
do Centro de Ciências e Tecnologia da Universi-
dade Federal do Cariri, como requisito parcial
à obtenção do grau de bacharel em Ciência da
Computação.

Orientadora: Profa. Dra. Luana Batista
da Cruz

JUAZEIRO DO NORTE - CE

2026

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da Universidade Federal do Cariri – UFCA

R756p Romão, Clíce Bezerra Brito.

Protótipo de sistema web para apoio à decisão clínica com classificação automatizada de anormalidades em exames gastrointestinais / Clíce Bezerra Brito Romão. – Juazeiro do Norte, CE, 2026.

97 f. : il. ; 30 cm.

Orientadora: Profa. Dra. Luana Batista da Cruz.

Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências e Tecnologia, Universidade Federal do Cariri, Juazeiro do Norte, CE, 2026.

1. 1. Diagnóstico auxiliado por computador. 2. Sistemas de apoio à decisão médica. 3. Endoscopia gastrointestinal — Processamento de imagem. 4. Inteligência artificial na medicina. 5. Sistemas de informação em saúde. I. Cruz, Luana Batista da, orient. II. Título.

CDD (22. ed.) 006.3

CLICE BEZERRA BRITO ROMÃO

PROTÓTIPO DE SISTEMA *WEB* PARA APOIO À DECISÃO CLÍNICA COM
CLASSIFICAÇÃO AUTOMATIZADA DE ANORMALIDADES EM EXAMES
GASTROINTESTINAIS

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Ciência da Computação
do Centro de Ciências e Tecnologia da Universi-
dade Federal do Cariri, como requisito parcial
à obtenção do grau de bacharel em Ciência da
Computação.

Aprovada em: 04/08/2026

BANCA EXAMINADORA

Profa. Dra. Luana Batista da Cruz (Orientadora)
Universidade Federal do Cariri (UFCA)

Prof. Dr. Nelson Carvalho Sandes
Universidade Federal do Cariri (UFCA)

Prof. Dr. João Otávio Bandeira Diniz
Instituto Federal de Educação, Ciência e Tecnologia
do Maranhão (IFMA)

AGRADECIMENTOS

À minha família, por todo o apoio, compreensão, incentivo e carinho ao longo desta caminhada acadêmica. Obrigada por acreditarem em mim e permanecerem ao meu lado em cada etapa deste processo, especialmente à minha mãe, Josete Bezerra Brito Romão, ao meu pai, Francisco Demontier Romão da Silva, às minhas avós, Maria Romão da Silva e Jucicleia Brito Bezerra, e ao meu avô, José Bezerra Lobo.

À minha professora orientadora Dra. Luana Batista da Cruz, por sua dedicação e paciência, pelos conhecimentos compartilhados ao longo do desenvolvimento do trabalho, por me orientar e esclarecer minhas dúvidas. Agradeço também à Universidade Federal do Cariri (UFCA), pela bolsa concedida durante minha participação no projeto LICA (Laboratório de Inteligência Computacional Aplicada), fundamental para a realização deste trabalho, e também pela oportunidade de aprendizado que surgiu em um momento inusitado, proporcionando a realização de um sonho.

Agradeço a todos os professores da UFCA pelos ensinamentos e pelas contribuições ao longo da minha formação. Aos colegas e amigos que fiz na universidade, agradeço pela amizade, pelo apoio e pelos momentos de descontração, que tornaram essa trajetória muito mais leve e enriquecedora, em especial à Nájla Maria Cavalcante dos Reis, Erika Natasha Nunes Ribeiro, João Victor, José Thalles Landim Silva, Gabrielly Xavier Alves Cruz e José Lucas Angelo dos Santos.

Aos meus amigos, Sabrina Santos Amorim, Raianne Ferreira Lima, Antonia Nirvana Gregorio Lima, José Weberth Brito Bezerra Jr. e Emerson Silva Biserra. À minha prima Hana Barros Bezerra Lobo de Brito e sua esposa, Letícia Marinho Perpétuo, pela paciência e pelo incentivo, e aos meus amigos de Brasília, que, apesar da distância, ainda estão muito presentes, Douglas Rodrigues do Amaral e Adrienne Martins Gomes de Albuquerque.

E por fim, meus agradecimentos a todos aqueles que, de alguma forma, contribuíram e me apoiaram ao longo desse percurso.

“I often tell my students not to be misled by the name ‘artificial intelligence’ – there is nothing artificial about it. AI is made by humans, intended to behave by humans, and, ultimately, to impact humans’ lives and human society.”

(Fei-Fei Li)

“Os livros não são feitos para acreditarmos neles, mas para serem submetidos a investigações.”

(Umberto Eco, *O Nome da Rosa*)

RESUMO

A análise automatizada de imagens endoscópicas por Inteligência Artificial tem se consolidado como uma ferramenta de apoio à identificação de anormalidades gastrointestinais. Entretanto, grande parte da literatura concentra-se em modelos de classificação isolados, sem integrá-los a plataformas de gestão clínica. Este estudo propõe um método que integra um módulo de *Computer-Aided Diagnosis*, baseado no método de Viana (2026) para a classificação de anormalidades de exames gastrointestinais, a um módulo de visualização implementado como um protótipo *web* de gestão clínica com *Role-Based Access Control*. O modelo de classificação reproduzido alcançou acurácia média de $95,97\% \pm 0,61\%$ e F1-score médio de $95,99\% \pm 0,60\%$, utilizando validação cruzada estratificada com $k = 3$ folds. A integração desses módulos resulta em um *Clinical Decision Support System* (CDSS), que disponibiliza aos médicos o resultado da classificação e mapas de ativação Grad-CAM, destacando visualmente as regiões da imagem que influenciaram a predição do modelo. Esses recursos auxiliam a avaliação clínica, enquanto o médico permanece responsável pelo julgamento diagnóstico e pela conclusão do exame. Os resultados obtidos indicam a viabilidade técnica dessa integração e estabelecem uma base para futuras validações em ambientes clínicos reais.

Palavras-chave: *Clinical Decision Support System*; *Computer-Aided Diagnosis*; Gestão Clínica; Imagens Endoscópicas; Inteligência Artificial;

ABSTRACT

The automated analysis of endoscopic images using Artificial Intelligence has increasingly been explored as a tool to support the identification of gastrointestinal abnormalities. However, much of the literature focuses on isolated classification models without integrating them into clinical management platforms. This study proposes a method that integrates a Computer-Aided Diagnosis module, based on the method by Viana (2026) for the classification of abnormalities in gastrointestinal exams, with a visualization module implemented as a web prototype for clinical management with Role-Based Access Control. The reproduced classification model achieved a mean accuracy of $95.97\% \pm 0.61\%$ and a mean F1-score of $95.99\% \pm 0.60\%$ using stratified 3-fold cross-validation. The integration of these modules results in a Clinical Decision Support System that provides physicians with the classification result and Grad-CAM activation maps, visually highlighting the image regions that influenced the model's prediction. These resources support clinical assessment, while the physician remains responsible for diagnostic judgment and the final interpretation of the examination. The results indicate the technical feasibility of this integration and provide a basis for future validation in real-world clinical settings.

Keywords: Artificial Intelligence; Clinical Decision Support System; Clinical Management; Computer-Aided Diagnosis; Endoscopic Images.

LISTA DE FIGURAS

Figura 1 – Exemplos de imagens endoscópicas digestiva alta e colonoscopia.	20
Figura 2 – Exemplo de imagem colonoscopia com <i>Specular Highlights</i> e <i>blurring</i>	21
Figura 3 – Exemplos de aplicação do Grad-CAM.	26
Figura 4 – Etapas do método proposto.	38
Figura 5 – Estrutura da base de imagens com separação de classes.	39
Figura 6 – Diagrama de casos de uso.	45
Figura 7 – Arquitetura geral do sistema.	46
Figura 8 – Diagrama de entidade-relacionamento do sistema	51
Figura 9 – Diagrama de estados do processamento do exame no sistema.	52
Figura 10 – Acurácia e F1-score do <i>Ensemble Stacking</i> por <i>fold</i>	55
Figura 11 – Tela de autenticação do sistema.	55
Figura 12 – Tela de <i>dashboard</i> do perfil ADM.	56
Figura 13 – Tela de <i>dashboard</i> do perfil GDC.	57
Figura 14 – Tela de <i>dashboard</i> do perfil MED.	57
Figura 15 – Tela de gerenciamento de papéis e permissões (RBAC).	58
Figura 16 – Tela de gerenciamento de usuários.	59
Figura 17 – Tela de <i>logs</i> de auditoria.	59
Figura 18 – Tela de cadastro de médicos (perfil GDC).	60
Figura 19 – Tela de cadastro de pacientes (perfil MED).	60
Figura 20 – Tela de cadastro de exame.	61
Figura 21 – Tela de exame “Pendente”.	61
Figura 22 – Tela de resultado da análise por IA.	62
Figura 23 – Relatório do exame.	63

LISTA DE TABELAS

Tabela 1 – Entidades do modelo de dados do sistema.	50
Tabela 2 – Desempenho médio e desvio padrão dos modelos e do <i>Ensemble Stacking</i> . .	54
Tabela 3 – Desempenho do <i>fold</i> 1.	54
Tabela 4 – Comparação do método proposto com trabalhos relacionados.	64
Tabela 5 – Requisitos funcionais de autenticação e controle de acesso.	76
Tabela 6 – Requisitos funcionais de gerenciamento de usuários.	77
Tabela 7 – Requisitos funcionais de gerenciamento de clínicas.	78
Tabela 8 – Requisitos funcionais de gerenciamento de pacientes.	79
Tabela 9 – Requisitos funcionais de gerenciamento de exames.	80
Tabela 10 – Requisitos funcionais de análise de exames com IA.	81
Tabela 11 – Requisitos funcionais de auditoria e rastreabilidade.	82
Tabela 12 – Requisitos funcionais de <i>dashboard</i> e monitoramento.	83
Tabela 13 – Requisitos não-funcionais de segurança.	84
Tabela 14 – Requisitos não-funcionais de persistência de dados.	85
Tabela 15 – Requisitos não-funcionais de modularização e organização arquitetural. . .	86
Tabela 16 – Requisitos não-funcionais de desempenho e confiabilidade.	87
Tabela 17 – Requisitos não-funcionais de escalabilidade.	88
Tabela 18 – Requisitos não-funcionais de portabilidade.	88
Tabela 19 – Requisitos não-funcionais de manutenibilidade.	89
Tabela 20 – Requisitos não-funcionais de usabilidade.	89
Tabela 21 – Regras de negócio de cadastro e vínculos de pacientes e exames.	90
Tabela 22 – Regras de negócio do fluxo de processamento de exames.	91
Tabela 23 – Regras de negócio de análise de IA.	92
Tabela 24 – Regras de negócio de segurança, auditoria e controle de acesso.	92
Tabela 25 – Casos de uso do sistema.	93
Tabela 26 – Casos de teste de autenticação e controle de acesso.	94
Tabela 27 – Casos de teste de usuários, clínicas e pacientes.	94
Tabela 28 – Casos de teste de gerenciamento de exames.	95
Tabela 29 – Casos de teste de análise de exames com IA.	95
Tabela 30 – Casos de teste de auditoria e rastreabilidade.	96
Tabela 31 – Casos de teste de infraestrutura e persistência.	96

LISTA DE ABREVIATURAS E SIGLAS

ACC	Acurácia
ADM	Administrator <i>Master</i>
AI	<i>Artificial Intelligence</i>
API	<i>Application Programming Interface</i>
CAD	<i>Computer-Aided Detection and Diagnosis</i>
CADe	<i>Computer-Aided Detection</i>
CADx	<i>Computer-Aided Diagnosis</i>
CDSSs	<i>Clinical Decision Support Systems</i>
CEP	Código de Endereçamento Postal
CNNs	<i>Convolutional Neural Networks</i>
CNPJ	Cadastro Nacional de Pessoas Jurídicas
CPF	Cadastro de Pessoas Físicas
CPU	<i>Central Processing Unit</i>
CTs	Casos de Teste
DL	<i>Deep Learning</i>
ESP	Especificidade
GDC	Gestor da Clínica
GPU	<i>Graphics Processing Unit</i>
Grad-CAM	<i>Gradient-weighted Class Activation Mapping</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IA	Inteligência Artificial
IARC	<i>International Agency for Research on Cancer</i>
IP	<i>Internet Protocol</i>
JSON	<i>JavaScript Object Notation</i>
JWT	<i>JSON Web Token</i>
LGPD	Lei Geral de Proteção de Dados
MED	Médico
ML	<i>Machine Learning</i>
ORM	<i>Object-Relational Mapping</i>
PDF	<i>Portable Document Format</i>
PRE	Precisão

RBAC	<i>Role-Based Access Control</i>
REST	<i>Representational State Transfer</i>
RFs	Requisitos Funcionais
RL	Regressão Logística
RNFs	Requisitos Não-Funcionais
RNs	Regras de Negócio
ROI	<i>Region of Interest</i>
SaaS	<i>Software as a Service</i>
SEN	Sensibilidade
SGBD	Sistema de Gerenciamento de Banco de Dados
SPAs	<i>Single-Page Applications</i>
SQL	<i>Structured Query Language</i>
UCs	<i>Use Cases</i>
UML	<i>Unified Modeling Language</i>
ViTs	<i>Vision Transformers</i>
XAI	<i>Explainable AI</i>

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Objetivo Geral	16
1.2	Objetivos Específicos	16
1.3	Organização do Trabalho	17
2	FUNDAMENTAÇÃO TEÓRICA	18
2.1	Doenças Gastrointestinais	18
2.2	Endoscopia Digestiva Alta e Colonoscopia	19
2.3	Inteligência Artificial na Gastroenterologia	22
2.3.1	<i>Machine Learning</i>	22
2.3.2	<i>Deep Learning</i>	23
2.3.2.1	<i>Convolutional Neural Networks</i>	23
2.3.2.2	<i>Vision Transformers</i>	24
2.3.3	<i>Ensemble Stacking</i>	24
2.3.4	<i>Explainable AI</i>	25
2.4	Sistemas Inteligentes Aplicados à Saúde	26
2.4.1	<i>Aspectos de Segurança e Privacidade em Sistemas de Saúde</i>	27
2.4.2	<i>Clinical Decision Support Systems</i>	28
2.5	Engenharia de Software e Padrões de Arquitetura Web	29
2.5.1	<i>Arquitetura Cliente-Servidor e Tecnologias de Interface</i>	30
2.5.1.1	<i>Biblioteca React e Desenvolvimento Baseado em Componentes</i>	30
2.5.1.2	<i>REST APIs e Framework FastAPI</i>	31
2.5.2	<i>Arquitetura em Camadas</i>	31
2.5.3	<i>Persistência de Dados com PostgreSQL</i>	32
2.5.4	<i>Migrations de Banco de Dados com Alembic</i>	32
2.5.5	<i>Virtualização de Containers com Docker</i>	32
2.5.6	<i>Versionamento com Git e Distribuição de Artefatos via GitHub Releases</i> . .	33
2.5.7	<i>Levantamento de Requisitos e Regras de Negócio</i>	33
2.5.8	<i>Diagramas de Casos de Uso</i>	34
2.5.9	<i>Máquina de Estados e Diagrama de Fluxo</i>	35
2.5.10	<i>Testes de Software</i>	35

3	TRABALHOS RELACIONADOS	36
3.1	Modelos CADx para Classificação de Imagens Gastrointestinais	36
3.2	CDSSs com Integração de Modelos CAD	36
4	MATERIAIS E MÉTODO PROPOSTO	38
4.1	Base de Imagens	39
4.2	Módulo de CADx	40
4.2.1	<i>Extração de ROI</i>	40
4.2.2	<i>Pré-Processamento</i>	40
4.2.3	<i>Ensemble Stacking</i>	40
4.2.4	<i>Métricas de Avaliação</i>	41
4.3	Módulo de Visualização	41
4.3.1	<i>Perfil do Administrador Master</i>	42
4.3.2	<i>Perfil do Médico</i>	42
4.3.3	<i>Perfil do Gestor da Clínica</i>	43
5	DESENVOLVIMENTO DO SISTEMA	44
5.1	Levantamento de Requisitos	44
5.2	Modelagem de Casos de Uso	44
5.3	Arquitetura do Sistema	46
5.3.1	<i>Camada Frontend</i>	46
5.3.2	<i>Camada Backend</i>	47
5.3.3	<i>Persistência dos Dados</i>	47
5.3.4	<i>Módulo de IA</i>	48
5.3.5	<i>Comunicação Entre os Componentes</i>	48
5.3.6	<i>Infraestrutura</i>	49
5.4	Modelagem do Banco de Dados	50
5.5	Fluxo de Processamento do Exame	52
6	RESULTADOS E DISCUSSÃO	53
6.1	Configuração Experimental	53
6.2	Resultados da Reprodução do Módulo de CADx	54
6.3	Resultados do Módulo de Visualização	55
6.4	Comparação com a Literatura	64
6.5	Testes Realizados	65

7	CONCLUSÕES E TRABALHOS FUTUROS	66
	REFERÊNCIAS	68
	APÊNDICES	76
	APÊNDICE A – Requisitos Funcionais	76
A.1	Autenticação e Controle de Acesso	76
A.2	Gerenciamento de Usuários	77
A.3	Gerenciamento de Clínicas	78
A.4	Gerenciamento de Pacientes	79
A.5	Gerenciamento de Exames	80
A.6	Análise de Exames com IA	81
A.7	Auditoria e Rastreabilidade	82
A.8	<i>Dashboard</i> e Monitoramento	83
	APÊNDICE B – Requisitos Não-Funcionais	84
B.1	Segurança	84
B.2	Persistência de Dados	85
B.3	Modularização e Organização Arquitetural	86
B.4	Desempenho e Confiabilidade	87
B.5	Escalabilidade	88
B.6	Portabilidade	88
B.7	Manutenibilidade	89
B.8	Usabilidade	89
	APÊNDICE C – Regras e Negócio	90
C.1	Cadastro e Vínculos de Pacientes e Exames	90
C.2	Fluxo de Processamento de Exames	91
C.3	Análise de IA	92
C.4	Segurança, Auditoria e Controle de Acesso	92
	APÊNDICE D – Casos de Uso	93
	APÊNDICE E – Casos de Teste	94
E.1	Autenticação e Controle de Acesso	94
E.2	Gerenciamento de Usuários, Clínicas e Pacientes	94
E.3	Gerenciamento de Exames	95
E.4	Análise de Exames com IA	95

E.5	Auditoria e Rastreabilidade	96
E.6	Infraestrutura e Persistência	96

1 INTRODUÇÃO

A transformação digital na saúde, impulsionada pelo avanço das Tecnologias da Informação e Comunicação, compreende a informatização de serviços e processos, a integração e o compartilhamento de informações, empregando tecnologias como telemedicina, prontuários eletrônicos e Inteligência Artificial (IA). Essas soluções podem contribuir para novas formas de atendimento, organização das informações clínicas e melhoria dos processos assistenciais (PINTO *et al.*, 2022; STOUUMPOS *et al.*, 2023). No entanto, sua adoção exige validação técnica e clínica, mecanismos de governança e medidas voltadas à privacidade, à segurança e à proteção dos dados de saúde (MATHEWS *et al.*, 2019).

O crescimento exponencial dos dados clínicos ampliou a necessidade de métodos capazes de processá-los e de extrair informações relevantes. Nesse contexto, técnicas de *Machine Learning* (ML) e *Deep Learning* (DL) passaram a ocupar papel importante na análise de imagens médicas, devido à sua capacidade de aprender padrões complexos em grandes volumes de dados (AWRAHMAN *et al.*, 2022; MOHAMMED *et al.*, 2024). Arquiteturas baseadas em *Convolutional Neural Networks* (CNNs), *Vision Transformers* (ViTs) ou modelos híbridos vêm sendo investigadas como abordagens para a representação de informações visuais em diferentes escalas (DIVYA; SUDHA, 2026).

A gastroenterologia está entre as áreas em que essas técnicas vêm sendo aplicadas. Doenças e neoplasias gastrointestinais causam grande impacto nos sistemas de saúde, e sua identificação precoce e precisa favorece tratamentos menos invasivos e melhores desfechos clínicos (FATI *et al.*, 2022; PATHAN *et al.*, 2026). Modelos de IA tem sido analisados e empregados para auxiliar na interpretação de exames gastrointestinais, apresentando resultados promissores na identificação automatizada, o que contribui na redução do tempo de análise dos exames (PATHAN *et al.*, 2026; DIVYA; SUDHA, 2026).

Esses modelos também podem ser incorporados a *Clinical Decision Support Systems* (CDSSs), ferramentas que fornecem informações, recomendações ou alertas, auxiliando profissionais na tomada de decisões clínicas, integrando dados de prontuários eletrônicos, bases de conhecimento, imagens médicas e algoritmos de análise (CHEN *et al.*, 2023). A incorporação de ML e DL ampliou as possibilidades dos CDSSs em tarefas como análise de imagens, predição de riscos e identificação de padrões, mas também introduziu desafios relacionados à integração aos fluxos clínicos, à interoperabilidade, à interpretabilidade e à aceitação pelos profissionais (ELHADDAD; HAMAM, 2024).

O desempenho preditivo desses modelos, contudo, não garante por si só sua aplicação segura ou sua aceitação na prática clínica. Muitos modelos de DL são caracterizados como *black boxes*, dificultando a compreensão dos critérios que influenciaram suas predições (SRINIVASU *et al.*, 2022). Técnicas de *Explainable AI* (XAI), como o *Gradient-weighted Class Activation Mapping* (Grad-CAM), procuram reduzir essa limitação por meio da geração de mapas que destacam regiões da imagem que levaram a classificação do modelo (SELVARAJU *et al.*, 2020). Esses mapas constituem recursos auxiliares de interpretação, mas não comprovam causalidade, localização correta da lesão ou validade clínica da predição.

Apesar dos avanços, a incorporação de modelos de IA aos fluxos clínicos ainda enfrenta desafios de integração, interoperabilidade, privacidade, segurança dos dados e aceitação dos profissionais (CHEN *et al.*, 2023), exigindo validação rigorosa antes da transposição de contextos experimentais para aplicações reais (EBAD *et al.*, 2025). Embora algumas soluções já tenham sido avaliadas em ambientes clínicos reais, grande parte delas permanece concentrada na avaliação isolada dos modelos, sem integração a uma plataforma de gestão clínica em que os resultados da IA sirvam apenas como apoio à decisão, permanecendo o profissional médico responsável pela conclusão do exame (NOPOUR, 2025; ARAÚJO; HORNUNG, 2022). Assim, o desenvolvimento de sistemas *web* integrados constitui uma alternativa para aproximar os avanços da IA das necessidades práticas das clínicas e profissionais de saúde.

1.1 Objetivo Geral

Desenvolver o ClinicAI, um protótipo de sistema *web* de gerenciamento clínico com suporte à classificação automatizada de exames gastrointestinais em normais ou anormais, utilizando IA como ferramenta de apoio à decisão médica.

1.2 Objetivos Específicos

- Realizar um levantamento bibliográfico para compreender o problema investigado e identificar possíveis lacunas;
- Implementar um módulo de gestão clínica com *Role-Based Access Control* (RBAC);
- Projetar uma arquitetura modular para a comunicação entre os componentes do sistema;
- Reproduzir um método de classificação automatizada de anormalidades gastrointestinais a partir da literatura;

- Integrar o modelo preditivo ao sistema, incorporando a classificação automatizada ao fluxo de processamento e revisão dos exames;
- Implementar um módulo de explicabilidade para auxiliar a interpretação dos resultados gerados pelo modelo preditivo;
- Adotar mecanismos de segurança, auditoria e rastreabilidade das operações realizadas no sistema;
- Avaliar o desempenho do modelo reproduzido em comparação com a literatura.

1.3 Organização do Trabalho

Os capítulos seguintes deste trabalho estão estruturados da seguinte forma:

- Capítulo 2 (Fundamentação Teórica): apresenta os fundamentos médicos, computacionais e de Engenharia de *Software* que sustentam o trabalho, desde conceitos clínicos às técnicas de IA e aos padrões de arquitetura *web* adotados;
- Capítulo 3 (Trabalhos Relacionados): apresenta e compara trabalhos relacionados à classificação de anormalidades em imagens gastrointestinais e à integração de modelos de IA, bem como estudos sobre o desenvolvimento de sistemas computacionais;
- Capítulo 4 (Materiais e Método Proposto): descreve o método experimental de reprodução do *Ensemble Stacking* proposto por Viana (2026) e a concepção do módulo de *Computer-Aided Diagnosis* (CADx) e do módulo de visualização que compõem o método proposto;
- Capítulo 5 (Desenvolvimento do Sistema): detalha o processo de Engenharia de *Software* conduzido para a implementação do protótipo;
- Capítulo 6 (Resultados e Discussão): apresenta e discute os resultados encontrados, comparando-os com a literatura;
- Capítulo 7 (Conclusões e Trabalhos Futuros): sintetiza as contribuições e os resultados do trabalho, relaciona-os aos objetivos estabelecidos, discute suas limitações e aponta direcionamentos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os fundamentos teóricos que sustentam o desenvolvimento deste trabalho. Inicialmente, são abordadas as doenças gastrointestinais e os principais exames diagnósticos de suas patologias: a endoscopia digestiva alta e a colonoscopia. Em seguida, são apresentados os conceitos de IA, ML e DL, e as principais técnicas de Visão Computacional empregadas na análise de imagens médicas, incluindo CNNs, ViTs, métodos de *Ensemble* e técnicas de XAI. Também são discutidos os sistemas inteligentes aplicados à saúde, com ênfase em segurança e privacidade de dados, os CDSSs e os padrões de interoperabilidade em saúde. Por fim, são apresentados os conceitos de Engenharia de *Software*, abrangendo elicitação de requisitos, modelagem de diagramas de estado e os padrões de arquitetura *web* que fundamentam a construção da solução proposta.

2.1 Doenças Gastrointestinais

O trato gastrointestinal humano é responsável pela ingestão, digestão e absorção de nutrientes, além da eliminação de resíduos metabólicos. Anatomicamente, é dividido em: superior, composto por cavidade oral, faringe, esôfago, estômago e duodeno; e inferior, formado pelo intestino delgado, cólon, reto e canal anal. Devido à sua extensa superfície mucosa e à exposição a fatores ambientais, alimentares, genéticos e microbiológicos, ele é suscetível ao desenvolvimento de diversas patologias ao longo da vida (MOORE *et al.*, 2019).

Do ponto de vista epidemiológico, as doenças gastrointestinais representam uma importante causa de morbidade e mortalidade em escala global, destacando-se os processos inflamatórios, infecções na mucosa, lesões ulcerativas, pólipos e neoplasias (FERLAY *et al.*, 2024). Apesar de apresentarem etiologias distintas, essas condições se manifestam por meio de alterações morfológicas na mucosa do trato digestório, que, em muitos casos, são silenciosas ou assintomáticas, dificultando seu diagnóstico em estágios iniciais (NIJJAR *et al.*, 2024).

Entre as enfermidades de maior relevância clínica estão, principalmente, a doença de Crohn e a retocolite ulcerativa, caracterizadas por inflamação crônica e recorrente da mucosa, avaliadas por meio de escores endoscópicos padronizados, embora a concordância entre observadores ainda represente um desafio na prática clínica (HASHASH *et al.*, 2024). Entre as neoplasias, merecem destaque os cânceres colorretal, gástrico e esofágico, que se destacam pela elevada incidência, mortalidade e impacto na saúde pública (FERLAY *et al.*, 2024).

O câncer colorretal é a neoplasia digestiva de maior incidência, sendo precedido, na maioria dos casos, pelos pólipos colorretais, lesões estruturais que evoluem lentamente ao longo de vários anos e que, por meio da sequência adenoma-carcinoma, podem se tornar malignos (SULLIVAN *et al.*, 2022). Segundo dados do GLOBOCAN 2022, divulgados pela *International Agency for Research on Cancer* (IARC), ele ocupa a terceira posição entre os cânceres mais diagnosticados do mundo, com mais de 1,9 milhão de novos casos, e a segunda posição em número de óbitos, com cerca de 900 mil mortes registradas no ano (FERLAY *et al.*, 2024).

Já o câncer gástrico ocupa a quinta posição tanto em incidência quanto em mortalidade entre os tipos de câncer no mundo, com cerca de 968 mil novos casos e 660 mil óbitos estimados em 2022 (FERLAY *et al.*, 2024). Assim como o câncer colorretal, seu diagnóstico costuma ser tardio, já que os sintomas iniciais tendem a ser inespecíficos e facilmente confundidos com condições benignas do trato digestório (NIJJAR *et al.*, 2024).

O câncer de esôfago, embora ocupe a décima primeira posição em incidência mundial, figura entre as sete principais causas de óbito por câncer, com cerca de 511 mil novos casos e 445 mil mortes estimadas em 2022 (FERLAY *et al.*, 2024). Essa desproporção entre incidência e mortalidade reflete o caráter agressivo da doença, sendo frequentemente diagnosticada apenas em estágios avançados (NIJJAR *et al.*, 2024).

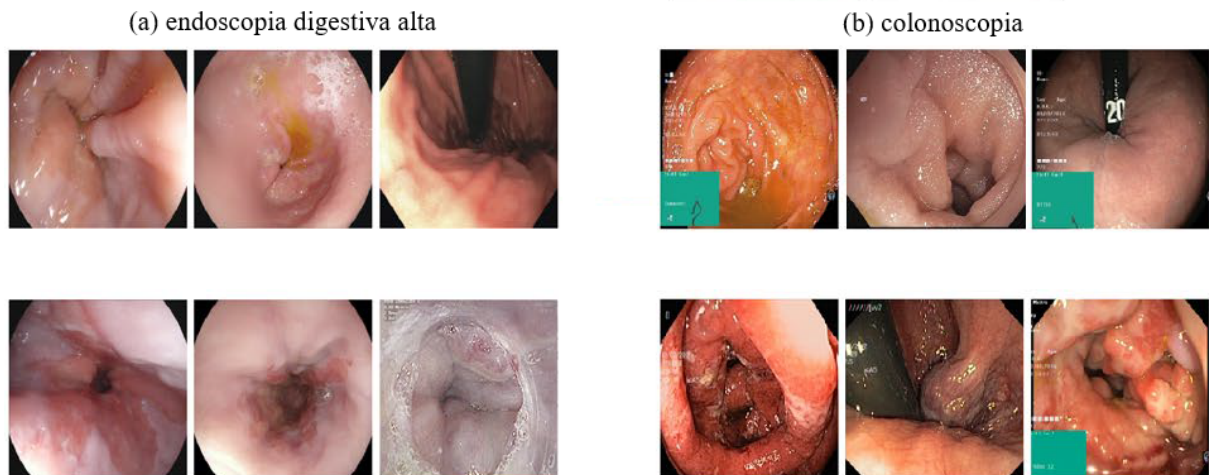
Dessa forma, a identificação precoce de alterações na mucosa constitui um dos principais desafios da gastroenterologia moderna, estando relacionada a um melhor prognóstico e taxas de sobrevida, além da redução da necessidade de tratamentos mais invasivos. Nesse cenário, exames de imagem que proporcionam a visualização direta do interior do trato gastrointestinal tornaram-se ferramentas indispensáveis para o rastreamento, diagnóstico e monitoramento dessas condições, destacando-se a endoscopia digestiva alta e a colonoscopia (SHAUKAT *et al.*, 2022; SOUZA *et al.*, 2024).

2.2 Endoscopia Digestiva Alta e Colonoscopia

No escopo da gastroenterologia moderna, a endoscopia digestiva alta e a colonoscopia consolidaram-se como os padrões-ouro para a avaliação, o diagnóstico e o tratamento de doenças no trato gastrointestinal. Esses exames são realizados com a introdução de um dispositivo endoscópico flexível, com uma câmera e um sistema de iluminação, que transmite imagens do interior do sistema digestório, permitindo a observação direta da mucosa e a identificação de alterações estruturais ou funcionais (SHAUKAT *et al.*, 2022; SOUZA *et al.*, 2024). A Figura 1

apresenta exemplos de imagens obtidas por (a) endoscopia digestiva alta e (b) colonoscopia.

Figura 1 – Exemplos de imagens endoscópicas digestiva alta e colonoscopia.



Fonte: Adaptado de (BORGLI *et al.*, 2020).

Cada exame possui foco anatômico específico. A endoscopia digestiva alta é voltada à inspeção do esôfago, do estômago e da parte inicial do duodeno, sendo essencial na identificação de doenças como esofagite de refluxo, úlceras e estágios iniciais do câncer gástrico (SOUZA *et al.*, 2024). Já a colonoscopia abrange todo o intestino grosso e a parte final do intestino delgado, permitindo identificar, delimitar e remover pólipos adenomatosos, interrompendo a sequência adenoma-carcinoma e reduzindo a mortalidade associada ao câncer colorretal (SHAUKAT *et al.*, 2022; SULLIVAN *et al.*, 2022).

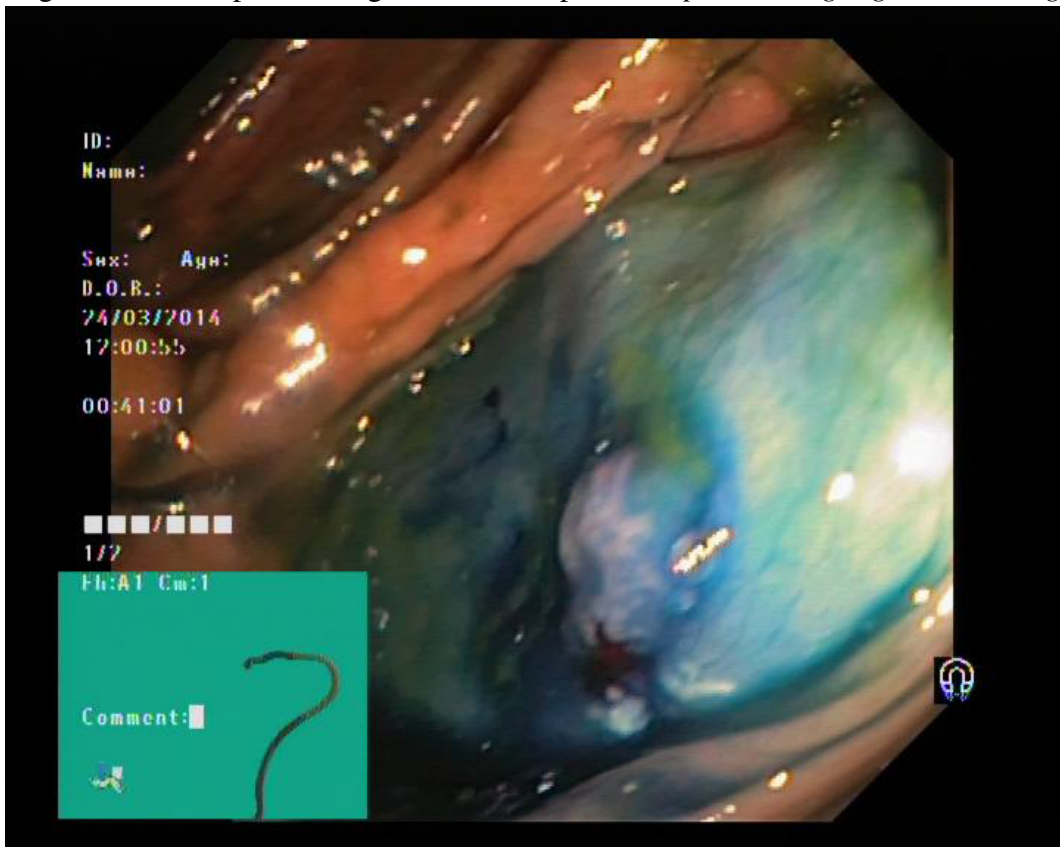
Apesar da alta relevância de ambos os métodos, a precisão diagnóstica permanece vinculada a fatores humanos e operacionais. Entre os fatores humanos, um dos desafios de maior impacto na prática clínica é a variabilidade interobservador e intraobservador, na qual diferentes médicos, ou o mesmo especialista, podem categorizar características de uma lesão de formas divergentes, impactando as decisões terapêuticas (HASHASH *et al.*, 2024; ASO *et al.*, 2025). Somados a isso, a fadiga cognitiva e a exaustão visual decorrentes de longas jornadas de trabalho também contribuem para o aumento de erros médicos, principalmente da taxa de adenomas não detectados (ASO *et al.*, 2025).

Entre os fatores operacionais, a eficácia clínica e a acurácia diagnóstica desses procedimentos são diretamente proporcionais à qualidade visual da mucosa obtida pelo examinador. No caso da colonoscopia, principalmente, o nível de limpeza intestinal é um indicador crítico de qualidade do exame, frequentemente quantificado por escalas validadas na literatura médica, já que um preparo inadequado impede a visualização das estruturas biológicas, elevando o risco

de oclusão de pólipos e lesões pré-cancerosas, comprometendo o sucesso técnico do método (BORGLI *et al.*, 2020; SOUZA *et al.*, 2024).

Há também restrições do ambiente gastrointestinal, como a umidade, que contribui para a oclusão parcial por fluidos biológicos, e os *Specular Highlights*, provocados pela incidência da luz direta sobre a superfície molhada. O peristaltismo (movimentos involuntários musculares) causam distorções, que, juntamente com as variações na distância entre a lente da câmera e a parede tecidual, geram baixo contraste ou desfoque (*blurring*), atuando como ruídos na formação da imagem (BORGLI *et al.*, 2020; FATI *et al.*, 2022). A Figura 2 ilustra uma imagem de colonoscopia afetada por *Specular Highlights* e *blurring*.

Figura 2 – Exemplo de imagem colonoscopia com *Specular Highlights* e *blurring*.



Fonte: Extraída da base de imagens *Kvasir VI* (POGORELOV *et al.*, 2017).

A combinação dessas restrições técnicas, operacionais e biológicas leva à subjetividade, o que torna vantajosa a introdução de arquiteturas de IA como uma segunda opinião contínua e imune à fadiga, atenuando ruídos e garantindo uma análise automatizada (ASO *et al.*, 2025; SHAUKAT *et al.*, 2022).

2.3 Inteligência Artificial na Gastroenterologia

Nas últimas décadas, a IA, ou *Artificial Intelligence* (AI), tem se consolidado como um pilar da saúde digital, impulsionada pelo crescimento da capacidade de processamento computacional e pela ampla disponibilidade de dados clínicos. De forma geral, corresponde à construção de sistemas computacionais capazes de simular processos cognitivos humanos, como o aprendizado de padrões, a adaptação a novos cenários e a resolução de problemas, evoluindo para arquiteturas capazes de analisar imagens médicas em tempo real (AKOLGO *et al.*, 2024).

Na gastroenterologia, essa incorporação ocorre principalmente por meio da Visão Computacional, subárea da IA voltada à interpretação de imagens, permitindo análises automatizadas a partir de sinais clínicos capturados durante os procedimentos (PATHAN *et al.*, 2026). Ela compreende três frentes principais: a classificação, voltada à categorização de imagens e de patologias; a detecção, aplicada à localização de anomalias; e a segmentação, responsável pela delimitação exata das bordas e da extensão da área afetada (DIVYA; SUDHA, 2026; MOHAMMED *et al.*, 2024). Essas frentes estão associadas à padronização das análises e à redução da variabilidade que compromete a interpretação de exames endoscópicos (HASHASH *et al.*, 2024; VAISHYA *et al.*, 2025) e vêm evoluindo de algoritmos clássicos de ML para arquiteturas de redes neurais profundas (KUFEL *et al.*, 2023; TALEBI *et al.*, 2026).

2.3.1 *Machine Learning*

O ML é um campo da IA focado no desenvolvimento de modelos computacionais capazes de identificar padrões complexos e extrair conhecimento de forma autônoma, sem a necessidade de regras programadas manualmente (KUFEL *et al.*, 2023). Na medicina moderna, os algoritmos clássicos de ML atuam como ferramentas de suporte à predição clínica, sobretudo no aprendizado supervisionado, em que o modelo é treinado com dados previamente rotulados por especialistas, estabelecendo correlações entre variáveis clínicas de entrada e um diagnóstico de referência (TALEBI *et al.*, 2026). Contudo, o emprego desses classificadores depende de uma etapa prévia manual e subjetiva de extração de atributos, chamada *feature engineering*, o que limita sua aplicação a imagens médicas complexas, como exames de endoscopia e colonoscopia, motivando a transição para arquiteturas de DL, capazes de aprender diretamente a partir de imagens brutas (MOHAMMED *et al.*, 2024; NOGUEIRA-RODRÍGUEZ *et al.*, 2021).

2.3.2 Deep Learning

O DL é um ramo do ML que se fundamenta na construção de arquiteturas com múltiplas camadas ocultas de processamento, que operam em níveis progressivos de abstração, emulando a organização das redes neurais biológicas (KUFEL *et al.*, 2023). A sua principal vantagem é a eliminação de *handcrafted features*, permitindo que a própria rede aprenda, de forma automatizada e hierárquica, as características ótimas para a solução de uma tarefa preditiva a partir de dados brutos (NOGUEIRA-RODRÍGUEZ *et al.*, 2021).

Essa capacidade de aprender representações diretamente dos dados torna o DL particularmente adequado para a análise de imagens endoscópicas, nas quais são frequentes variações de iluminação, reflexos na mucosa e oclusões por fluidos digestivos (FATI *et al.*, 2022; MOHAMMED *et al.*, 2024). Nesse contexto, destacam-se as CNNs e os ViTs, arquiteturas amplamente empregadas em Visão Computacional para análise de imagens médicas, contribuindo para a detecção automatizada de anormalidades gastrointestinais (DIVYA; SUDHA, 2026; WU *et al.*, 2024).

2.3.2.1 Convolutional Neural Networks

No âmbito da medicina diagnóstica, as CNNs, ou Redes Neurais Convolucionais, consolidaram-se como a arquitetura padrão-ouro para as tarefas de classificação, detecção e segmentação de imagens médicas (MOHAMMED *et al.*, 2024). Diferentemente das redes neurais tradicionais, as CNNs utilizam a operação matemática de convolução, aplicando filtros lineares que deslizam sobre a matriz de *pixels* da imagem para extrair, de forma progressiva, características primitivas e complexas, tais como bordas, texturas, formas geométricas e variações sutis na coloração da mucosa digestiva (KUFEL *et al.*, 2023; MOHAMMED *et al.*, 2024).

Dentre as topologias mais relevantes para a gastroenterologia estão a ResNet-50, que introduziu conexões residuais para amenizar o problema do desvanecimento do gradiente em redes muito profundas, viabilizando o treinamento estável de modelos complexos e o reaproveitamento de modelos já treinados (*Transfer Learning*) para o mapeamento de anomalias na mucosa digestiva (AYAN, 2024; GELU-SIMEON *et al.*, 2025); e a EfficientNet-B4, que otimiza a relação entre acurácia e custo computacional por meio do escalonamento composto, consolidando-se como uma solução viável para o processamento médico em tempo real (GANGRADE *et al.*, 2025; TAN; LE, 2019).

2.3.2.2 Vision Transformers

Apesar do sucesso consolidado das CNNs, os ViTs emergiram como uma alternativa robusta, adaptando os mecanismos de *self-attention* ao domínio das imagens (DIVYA; SUDHA, 2026). Diferentemente das CNNs, que processam a informação local de *pixels*, os ViTs segmentam a imagem em uma sequência de blocos bidimensionais, chamados *patches*, avaliando por meio da *self-attention* a correlação contextual global entre eles (MAURÍCIO *et al.*, 2023).

Um exemplo da aplicação dessa tecnologia ao rastreamento colorretal é a arquitetura PVTv2-B2, que introduz uma estrutura piramidal capaz de gerar mapas de características multiescala, superando a limitação de resolução fixa e o alto custo computacional dos ViTs convencionais (WANG *et al.*, 2022). Ao reduzir a complexidade da *self-attention* por meio de aproximações lineares, essa topologia otimiza a segmentação de limites tumorais complexos, pólipos planos e lesões inflamatórias milimétricas frequentemente despercebidas em triagens convencionais (DONG *et al.*, 2023).

2.3.3 Ensemble Stacking

A consolidação de diferentes modelos de DL abre caminho para abordagens híbridas que superam as limitações intrínsecas de cada arquitetura computacional isolada. A técnica de *Ensemble Stacking* atua como uma estratégia avançada de aprendizagem combinada, integrando múltiplos modelos de base heterogêneos para gerar um meta-modelo unificado de desempenho superior, permitindo a potencialização das virtudes e redução das fraquezas dos modelos isolados (SIVARI *et al.*, 2023; VIANA *et al.*, 2026).

O fluxo de processamento do *Ensemble Stacking* é estruturado em dois níveis. No primeiro (Nível 0), os modelos preditivos especializados processam as imagens brutas e geram saídas independentes, expressas como matrizes de probabilidade (ASLAN, 2025). No segundo (Nível 1), um meta-classificador recebe essas previsões como novos vetores de características de entrada, chamados *meta-features*, aprendendo a ponderar os pontos fortes e a corrigir os vieses estatísticos sistemáticos de cada classificador isolado (GANGRADE *et al.*, 2025). Por se tratar de uma etapa de combinação, e não de extração de características, o meta-classificador costuma ser um modelo mais simples que os classificadores de base, sendo a Regressão Logística (RL), uma escolha amplamente adotada na literatura por seu baixo custo computacional e boa interpretabilidade (SIVARI *et al.*, 2023; ASLAN, 2025).

2.3.4 Explainable AI

Apesar dos avanços obtidos pelos modelos de DL no diagnóstico por imagem, a sua aplicação enfrenta uma barreira crítica, conhecida na literatura como *black boxes* (CHADDAD *et al.*, 2023; MUHAMMAD; BENDECHACHE, 2024). Apesar da elevada precisão estatística ao classificar uma patologia, esses algoritmos não fornecem justificativas compreensíveis sobre como chegam a uma decisão (CHADDAD *et al.*, 2023; SRINIVASU *et al.*, 2022). Neste cenário, a XAI assume o papel de projetar modelos que permitam aos profissionais de saúde auditar, rastrear e compreender os critérios adotados na formulação de um diagnóstico assistido (SALIMPARSA *et al.*, 2025), proporcionando legitimidade clínica ao sistema, já que um médico dificilmente baseará uma conduta terapêutica ou cirúrgica em uma recomendação cuja fundamentação lógica seja desconhecida (CHADDAD *et al.*, 2023).

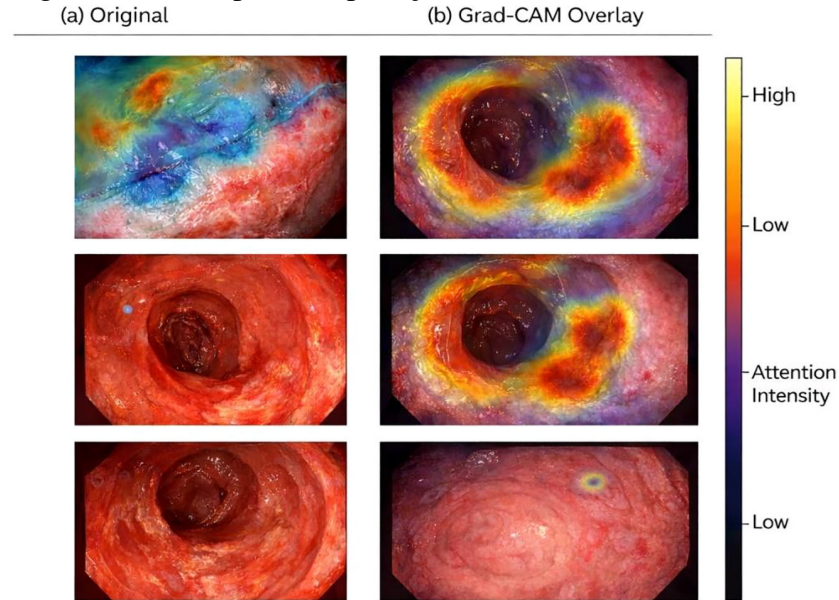
Do ponto de vista técnico, a disponibilização de explicações visuais permite ao especialista identificar quando o modelo baseia sua decisão em regiões da imagem sem relevância clínica, evidenciando desde falhas operacionais até falsas correlações aprendidas pela rede, antes que resultem em negligência diagnóstica ou danos reais ao paciente (PATHAN *et al.*, 2026; VAISHYA *et al.*, 2025). Já do ponto de vista regulatório, essa mesma capacidade de explicação atende demandas de governança em saúde digital, entre as quais está o “direito à explicação” diante de decisões automatizadas (VELDEN *et al.*, 2022). Esse preceito assegura que pacientes e médicos tenham acesso a explicações claras sobre triagens e diagnósticos automatizados, garantindo a equidade, a proteção de dados e o respaldo legal (ARAÚJO; HORNUNG, 2022; MENNELLA *et al.*, 2024).

Como reflexo prático dos conceitos de XAI, surge o método Grad-CAM, uma técnica de atribuição pós-treino amplamente utilizada para conter a opacidade de redes profundas. O algoritmo calcula o gradiente da pontuação obtida para uma determinada classe em relação aos mapas de ativação da última camada convolucional da rede, ponderando a importância de cada um deles e aplicando uma função de ativação *Rectified Linear Unit*, mantendo apenas as contribuições que favorecem positivamente a classe de interesse (SELVARAJU *et al.*, 2020). O resultado é um mapa de calor visual, projetado sobre a imagem original, em que áreas em tons quentes, como vermelho e amarelo, indicam onde a rede concentrou sua atenção, enquanto áreas em tons frios (azul e verde) indicam estruturas ignoradas pelo algoritmo.

Essa técnica permite ao médico verificar se a região destacada é coerente com a lesão ou se o modelo foi induzido ao erro por artefatos visuais no exame, contribuindo para que

CDSSs possam ser adotados com maior confiança pelos profissionais de saúde (ZHEN *et al.*, 2026). A Figura 3 apresenta (a) imagens originais e a (b) camada do Grad-CAM em imagens de colonoscopia.

Figura 3 – Exemplos de aplicação do Grad-CAM.



Fonte: Adaptado de Zhen *et al.* (2026).

2.4 Sistemas Inteligentes Aplicados à Saúde

A crescente digitalização dos serviços de saúde tem impulsionado a adoção de sistemas inteligentes capazes de auxiliar profissionais na coleta, no processamento e na interpretação de informações clínicas. Esse cenário está diretamente associado à expansão dos prontuários eletrônicos, sistemas de informação hospitalar e plataformas de telemedicina, favorecendo novas formas de organização dos fluxos assistenciais (STOUMPOS *et al.*, 2023).

Dessa forma, a saúde digital consolidou-se como um campo interdisciplinar que integra tecnologias computacionais às atividades assistenciais, administrativas e gerenciais, contribuindo para a interoperabilidade entre ambientes distintos, para a melhoria da qualidade assistencial e da segurança do paciente (PINTO *et al.*, 2022; STOUMPOS *et al.*, 2023). Contudo, esse crescimento nem sempre é acompanhado de validação técnica, clínica e de usabilidade rigorosa, o que reforça a importância de projetar esses sistemas com requisitos bem definidos e critérios claros de qualidade (MATHEWS *et al.*, 2019).

Entre essas tecnologias, destacam-se os métodos de IA descritos na Seção 2.3, empregados em sistemas de saúde para tarefas como classificação, detecção e diagnóstico de

anomalias (STOUMPOS *et al.*, 2023). Para que soluções possam ser adotadas de forma segura em ambientes clínicos reais, no entanto, é fundamental observar aspectos de segurança, privacidade dos dados e interoperabilidade, além dos CDSSs que integram essas tecnologias.

2.4.1 Aspectos de Segurança e Privacidade em Sistemas de Saúde

A proteção e a governança das informações em ambientes de saúde fundamentam-se na manutenção dos princípios da Segurança da Informação, sendo eles confidencialidade, integridade e disponibilidade. A confidencialidade assegura que dados clínicos sensíveis sejam acessados única e exclusivamente por profissionais devidamente autorizados; a integridade garante que o histórico de exames e laudos permaneça correto, livre de corrupções ou modificações maliciosas; enquanto a disponibilidade assegura o acesso à informação sempre que demandado para apoiar decisões médicas, atividades assistenciais e administrativas (NANKYA *et al.*, 2024).

No cenário brasileiro, a implementação de sistemas digitais de saúde deve alinhar-se aos preceitos da Lei Geral de Proteção de Dados (LGPD), que tipifica dados de saúde, relatórios clínicos e exames de imagem estruturados como dados sensíveis (Brasil, 2018). Consequentemente, as instituições devem adotar mecanismos técnicos e administrativos que assegurem a proteção dessas informações ao longo de todo o ciclo de vida, da coleta ao descarte (Brasil, 2018; PINTO *et al.*, 2022), destacando-se, entre eles, a criptografia de dados, a autenticação de usuários, o registro de auditoria e o controle de acesso (BUTPHENG *et al.*, 2020).

A criptografia protege os dados sensíveis tanto em trânsito quanto em repouso, tornando-os ilegíveis a terceiros não autorizados mesmo em caso de interceptação de rede ou acesso indevido ao armazenamento, sendo especialmente relevante para dados de saúde, cuja exposição pode comprometer a privacidade e a segurança do paciente. Já a autenticação verifica a identidade de quem acessa o sistema antes de conceder qualquer permissão, geralmente por meio de credenciais de acesso combinadas a mecanismos de controle de sessão, como *tokens* temporários que expiram após um período determinado, reduzindo o risco de uso indevido em caso de comprometimento das credenciais (BUTPHENG *et al.*, 2020).

Os registros (*logs*) de auditoria documentam operações como acessos, alterações e exclusões de dados, permitindo identificar responsáveis e reconstruir o histórico de eventos em caso de falhas ou incidentes de segurança, apoiando o princípio da responsabilidade previsto na LGPD (NANKYA *et al.*, 2024). Já, o controle de acesso é implementado, na prática, por meio do modelo de RBAC, ou Controle de Acesso Baseado em Papéis, no qual os privilégios de

navegação e alteração são vinculados não a um indivíduo, mas às funções organizacionais que ele ocupa, organizando-se em torno de usuários, papéis e permissões (BUTT *et al.*, 2023).

Essa última abordagem associa-se ao princípio de segurança do menor privilégio, segundo o qual cada agente acessa exclusivamente as informações e rotinas necessárias às suas funções. Em uma plataforma de apoio diagnóstico gastroenterológico, por exemplo, o corpo administrativo retém acesso apenas aos metadados cadastrais dos exames, enquanto o processamento inteligente, a visualização dos resultados e a validação de laudos permanecem restritos a especialistas cadastrados e autenticados, o que blinda o sistema contra vazamentos e garante conformidade legal (BUTT *et al.*, 2023).

Por fim, sistemas que atendem múltiplas organizações costumam adotar também o isolamento de dados por meio de arquiteturas *multi-tenant*. Esse padrão é amplamente adotado em soluções de *Software as a Service* (SaaS), no qual múltiplas organizações, chamadas *tenants*, compartilham a mesma aplicação e base de dados, ao mesmo tempo que cada uma permanece isolada. Esse isolamento determina a quais registros, pertencentes a qual organização, esse usuário pode ter acesso. Em sistemas que lidam com dados sensíveis, como prontuários e laudos médicos, essa verificação deve ser aplicada de forma sistemática na camada de acesso a dados, e uma falha nesse isolamento deve ser tratada como uma vulnerabilidade de segurança, não apenas como um erro funcional (BEZEMER; ZAIDMAN, 2010).

2.4.2 Clinical Decision Support Systems

Os CDSSs caracterizam-se como plataformas de *software* desenvolvidas com o propósito de assistir à equipe médica durante a execução de processos diagnósticos, terapêuticos e de monitoramento de pacientes, processando dados de prontuários eletrônicos, exames laboratoriais e de imagem para emitir alertas, predições e recomendações, com o objetivo de contribuir para decisões mais seguras e assertivas (CHEN *et al.*, 2023; ELHADDAD; HAMAM, 2024).

Os avanços recentes em DL transformaram os CDSSs tradicionais em ferramentas capazes de identificar padrões complexos em tempo real, produzindo recomendações mais contextualizadas às condições específicas dos pacientes (CHEN *et al.*, 2023; ELHADDAD; HAMAM, 2024). No contexto específico do diagnóstico por imagem, os sistemas *Computer-Aided Detection and Diagnosis* (CAD) constituem uma subcategoria especializada de CDSSs, subdividida em *Computer-Aided Detection* (CADE) e CADx.

Os sistemas CADE são voltados à detecção e delimitação de potenciais anormalidades

na imagem, enquanto os CADx focam na classificação e inferência sobre a natureza patológica das alterações já detectadas. Atuando de forma conjunta, esses dois sistemas funcionam como filtros de atenção visual, reduzindo a probabilidade de falsos negativos induzidos pela fadiga, distração ou sobrecarga cognitiva humana (PATHAN *et al.*, 2026; TALEBI *et al.*, 2026).

Técnicas de explicabilidade, como o Grad-CAM, podem ser incorporadas tanto a sistemas CAdE quanto a CADx, auxiliando na interpretação dos resultados gerados. Em um estudo clínico multicêntrico e randomizado, Shaukat *et al.* (2022) demonstraram que a introdução de um dispositivo CAdE elevou significativamente a taxa de detecção de adenomas durante colonoscopias convencionais, reforçando a eficácia clínica dessas ferramentas em ambientes médicos reais.

Apesar do elevado desempenho demonstrado pelas soluções baseadas em IA, esses sistemas devem ser compreendidos como ferramentas acessórias e complementares à medicina, já que suas predições, métricas e mapas de calor não possuem confirmação diagnóstica nem substituem a avaliação médica individualizada, permanecendo a decisão final como dever exclusivo do profissional. Os CDSSs atuam expandindo a capacidade analítica dos especialistas no ponto de cuidado, sem interferir em sua autonomia clínica ou em sua responsabilidade ética, civil e legal perante o paciente (ARAÚJO; HORNUNG, 2022; CHEN *et al.*, 2023).

2.5 Engenharia de Software e Padrões de Arquitetura Web

A materialização prática de CDSSs baseados em IA exige uma infraestrutura robusta, escalável e segura de Tecnologia da Informação. No ambiente hospitalar, a transição de modelos teóricos de ML para ferramentas operacionais integradas à rotina assistencial depende, fundamentalmente, de princípios consolidados da Engenharia de Software (ALEXIUK *et al.*, 2024). O projeto arquitetural de sistemas médicos deve limitar gargalos de processamento e garantir atributos essenciais de qualidade de *software*, tais como disponibilidade, manutenibilidade, interoperabilidade e portabilidade, estruturando de forma sistemática as camadas que gerenciam a interface do usuário, o tráfego de dados e o processamento inteligente das informações clínicas (PRESSMAN; MAXIM, 2016; TORAB-MIANDOAB *et al.*, 2023).

2.5.1 Arquitetura Cliente-Servidor e Tecnologias de Interface

O modelo de Arquitetura Cliente-Servidor é amplamente utilizado para organizar sistemas *web*, pois permite separar as responsabilidades entre o componente de interação com o usuário e o de processamento de informações (SAMPAIO, 2022). O componente Cliente corresponde à interface gráfica com a qual o usuário interage diretamente com as funcionalidades (PRESSMAN; MAXIM, 2016), enquanto o Servidor isola a infraestrutura lógica e os recursos de alta performance, assumindo a carga computacional de processamento e as consultas ao banco de dados (JIANG *et al.*, 2023).

No caso de um sistema de saúde digital, o Cliente concentra funcionalidades ligadas ao ponto de cuidado de pacientes, como o gerenciamento de laudos e a exibição dos resultados dos exames, buscando garantir usabilidade e uma experiência fluida para o especialista (SAMPALIO, 2022). Já o Servidor concentra as tarefas computacionalmente mais pesadas, como as inferências de redes neurais sobre imagens endoscópicas, isolando-as da interface do usuário. Esse isolamento entre as camadas é crítico no ambiente médico, assegurando a modularidade e o desempenho do sistema, impedindo que tarefas complexas comprometam sua execução contínua (SAMPALIO, 2022; JIANG *et al.*, 2023).

Para a aplicação prática dessa arquitetura, destacam-se ferramentas específicas para cada camada. Para o Cliente, destaca-se a biblioteca React, voltada ao desenvolvimento de interfaces dinâmicas baseadas em componentes reutilizáveis e ao gerenciamento de estados em *Single-Page Applications* (SPAs) (Meta Open Source, 2026). Já para o Servidor, destaca-se o *framework* FastAPI, baseado em Python e voltado à construção de servidores de alta performance capazes de gerenciar requisições assíncronas (RAMÍREZ, 2018).

2.5.1.1 Biblioteca React e Desenvolvimento Baseado em Componentes

O desenvolvimento baseado em componentes constitui um dos principais paradigmas empregados na construção de interfaces *web* modernas. Nesse contexto, o React implementa esse modelo ao organizar a interface em blocos reutilizáveis, independentes e autocontidos, favorecendo a modularização, a reutilização de código e o isolamento das responsabilidades da interface (Meta Open Source, 2026; CHHETRI, 2024).

Estudos indicam que a modularização reduz redundâncias no código-fonte e os custos de manutenção, além de acelerar os ciclos de testes e a integração com novas funcionalidades

(CHHETRI, 2024; PRESSMAN; MAXIM, 2016). Em sistemas com múltiplos perfis de acesso, essa abordagem permite reaproveitar componentes de interface entre diferentes telas e papéis de usuário, reduzindo a duplicação de código e facilitando a manutenção de funcionalidades específicas de cada perfil.

Em SPAs, a ausência de recarregamento entre telas exige que o estado da aplicação seja mantido e sincronizado inteiramente pela camada Cliente. Como não há uma solução única para esse caso, a literatura recomenda avaliar a necessidade de usar bibliotecas mais robustas frente aos mecanismos nativos do React (como *Context API* e *hooks*), à medida que a interface cresce em número de telas e fluxos de dados compartilhados (DROŃ'; SZANDAŁA, 2026). Em sistemas de análise médica, por exemplo, o *status* de um exame precisa ser consistente entre a submissão, a revisão médica e o *dashboard*, o que demanda essa sincronização.

2.5.1.2 REST APIs e Framework FastAPI

No desenvolvimento de sistemas modernos voltados à área da saúde e IA, a escolha do *framework* de suporte no *backend* é crítica para amenizar gargalos operacionais causados pelo alto fluxo de requisições e processamento de dados médicos. Ao adotar o padrão arquitetural *Representational State Transfer* (REST) para a construção de uma *Application Programming Interface* (API), é possível estabelecer a comunicação entre a camada de visualização e a de processamento por meio de rotas e do protocolo *Hypertext Transfer Protocol* (HTTP) (como GET e POST), trafegando requisições estruturadas que encapsulam dados clínicos e metadados em formato *JavaScript Object Notation* (JSON) (ODOFIN *et al.*, 2023). No FastAPI, o processamento dessas requisições ocorre de forma assíncrona, permitindo que o servidor atenda a múltiplas requisições simultâneas sem bloquear a execução, o que otimiza a taxa de transferência e reduz a latência em cenários de alta concorrência (RAMÍREZ, 2018; ALA'ANZY; YESHPATOV, 2026).

2.5.2 Arquitetura em Camadas

Além da separação entre Cliente e Servidor, a organização interna de cada componente costuma seguir o padrão de Arquitetura em Camadas, que consiste em estruturar o sistema em níveis hierárquicos com responsabilidades bem delimitadas, de modo que cada camada se comunique apenas com as adjacentes, favorecendo a manutenibilidade e a evolução independente do sistema (PRESSMAN; MAXIM, 2016; SAMPAIO, 2022). Normalmente, esse modelo é composto pela camada de apresentação, responsável pela interface com o usuário, pela camada

lógica, onde são definidas as regras e os processos centrais da aplicação, e pela camada de dados, encarregada da persistência e da recuperação das informações (PRESSMAN; MAXIM, 2016).

2.5.3 Persistência de Dados com PostgreSQL

A confiabilidade de uma aplicação depende diretamente da escolha de um Sistema de Gerenciamento de Banco de Dados (SGBD) capaz de assegurar integridade estrutural e persistência robusta (REDDYKAVULURI, 2024). O PostgreSQL destaca-se como um SGBD Objeto-Relacional de código aberto, cuja conformidade com as propriedades ACID (Atomicidade, Consistência, Isolamento e Durabilidade) garante a execução segura de transações e protege o sistema contra falhas ou corrupção (The PostgreSQL Global Development Group, 2025; WENG; WU, 2025), favorecendo ainda a modelagem estruturada de dados clínicos e a execução de consultas complexas sobre grandes volumes de registros, características relevantes para sistemas de saúde digital (URNIKIENÉ *et al.*, 2026; ALI *et al.*, 2025).

2.5.4 Migrations de Banco de Dados com Alembic

Ao longo do ciclo de vida de uma aplicação, o banco de dados relacional também tende a evoluir para acomodar novas funcionalidades, exigindo a criação, alteração ou remoção de tabelas, colunas e campos. Segundo Ambler e Sadalage (2006), essas mudanças devem ser tratadas como refatorações incrementais e reversíveis, aplicadas de forma controlada, de modo a preservar a integridade dos dados já existentes e a permitir que o mesmo *schema* seja reproduzido de forma consistente em diferentes ambientes.

O Alembic operacionaliza esses princípios por meio de *migrations*, nos quais cada alteração estrutural do banco de dados é registrada como um *script* de versão vinculado ao anterior, formando um histórico rastreável do *schema*. Cada *script* define tanto os passos de atualização (*upgrade*) quanto os de reversão (*downgrade*), permitindo que essas mudanças sejam aplicadas de forma automatizada e consistente entre diferentes ambientes (Alembic, 2026).

2.5.5 Virtualização de Containers com Docker

O processo de implantação de aplicações modernas enfrenta desafios clássicos associados à incompatibilidade de dependências de *software* e divergências entre os ambientes de desenvolvimento e produção (OPENJA, 2022). Para contornar essas barreiras, adota-se

a virtualização por *containers*, que permite empacotar uma aplicação juntamente com suas dependências em uma unidade isolada e portátil. O Docker é a principal plataforma que viabiliza a construção e a execução padronizada de *containers* (Docker Inc., 2026).

Diferentemente das máquinas virtuais tradicionais, que exigem a emulação de um sistema operacional, os *containers* Docker são mais leves por compartilharem o núcleo do sistema hospedeiro, reduzindo o consumo de recursos e o tempo de inicialização (SILVA *et al.*, 2024). Essa característica garante que o código e suas dependências operem de maneira idêntica e reproduzível em qualquer ambiente, otimizando a distribuição e a escalabilidade de microserviços (OPENJA, 2022; MARELLA, 2024).

2.5.6 Versionamento com Git e Distribuição de Artefatos via GitHub Releases

Um *Distributed Version Control System* registra o histórico de alterações de um projeto de *software*, possibilitando que vários desenvolvedores colaborem sem sobrescrever o trabalho uns dos outros. O Git é um dos sistemas desse tipo mais utilizados atualmente, organizando o histórico do projeto em *commits* (estados específicos dos arquivos) e permitindo a criação de *branches* para o desenvolvimento paralelo de funcionalidades. O GitHub é uma das plataformas de hospedagem de repositórios Git mais utilizadas, agregando funcionalidades de colaboração, como revisão de código por meio de *pull requests* e rastreamento de problemas (*issues*) (CHACON; STRAUB, 2014).

Os repositórios Git, no entanto, não são adequados para o versionamento de arquivos binários volumosos, como pesos de modelos de DL, já que cada alteração nesses arquivos é armazenada por completo no histórico. O GitHub Releases contorna essa limitação ao associar um conjunto de arquivos binários a uma *tag* específica do repositório, distribuindo artefatos sem a necessidade de versioná-los no próprio projeto, além de viabilizar a verificação de integridade desses arquivos por meio de *hashes* criptográficos antes de sua utilização (GitHub, 2026).

2.5.7 Levantamento de Requisitos e Regras de Negócio

O levantamento de requisitos constitui uma das etapas iniciais e mais críticas do processo de Engenharia de *Software*, responsável por definir o escopo, as funcionalidades e as restrições que orientam o desenvolvimento de um sistema (SOMMERVILLE, 2011), sendo dividido em duas categorias complementares:

- Requisitos Funcionais (RFs): descrevem o que o sistema deve fazer e, dependendo de

quem o utiliza, como deve reagir e se comportar em determinadas situações, podendo também explicar o que o sistema não deve fazer (SOMMERVILLE, 2011);

- Requisitos Não-Funcionais (RNFs): relacionados às propriedades do sistema, servindo como características que influenciam seu funcionamento como um todo, como segurança, desempenho, escalabilidade, usabilidade, entre outros (SOMMERVILLE, 2011).

Sistemas voltados a domínios com forte componente regulatório, como o de gestão clínica, também exigem a formalização de Regras de Negócio (RNs). Segundo Ross (2005), as RNs são declarações que definem ou restringem aspectos de negócio, assegurando que o comportamento do sistema esteja em conformidade com políticas, condições ou regulamentações estabelecidas. Diferentemente dos requisitos funcionais, elas expressam políticas do domínio de aplicação, frequentemente vinculadas a normas éticas, legais ou institucionais.

A elicitação de requisitos corresponde à etapa em que os requisitos do sistema são derivados a partir de diferentes fontes de informação, como sistemas similares já em uso, conversas com os futuros usuários e o estudo das atividades que o sistema deverá apoiar. Como parte dessa investigação, é comum a construção de protótipos do sistema, que contribuem para o entendimento do problema a ser resolvido, oferecendo um meio de usuários interagirem com o sistema, contribuindo para a elicitação e validação dos requisitos (SOMMERVILLE, 2011).

2.5.8 Diagramas de Casos de Uso

A partir dos RFs levantados, é comum representá-los através de diagramas de casos de uso (*Use Cases* (UCs)), um dos modelos comportamentais definidos pela *Unified Modeling Language* (UML) (SOMMERVILLE, 2011). Esse tipo de diagrama descreve as interações entre os atores, entidades externas que interagem com o sistema, podendo ser usuários humanos ou outros sistemas, e os UCs, unidades de funcionalidade que representam um objetivo que o ator alcança ao interagir com o sistema (SOMMERVILLE, 2011).

Além da associação direta entre um ator e um UC, a notação também permite representar relações entre os próprios UCs. Uma delas é a relação de inclusão (*include*), na qual um UC incorpora, obrigatoriamente, o comportamento de outro durante sua execução, geralmente empregada para representar uma funcionalidade comum reutilizada por diferentes fluxos do sistema (SOMMERVILLE, 2011). Assim, os diagramas de casos de uso funcionam como uma ponte entre o levantamento de requisitos e o projeto do sistema, oferecendo uma visão de quem interage com a aplicação e de quais funcionalidades cada perfil de usuário pode acessar.

2.5.9 Máquina de Estados e Diagrama de Fluxo

Além dos modelos estruturais, também são empregados modelos comportamentais para representar como um sistema reage a eventos internos e externos ao longo do tempo. Entre esses modelos, a UML adota a abordagem de modelagem orientada a eventos por meio dos diagramas de estado, que representam os estados possíveis de um sistema e os eventos responsáveis por provocar transições (SOMMERVILLE, 2011).

Nessa notação, os estados são representados por retângulos arredondados, com breves descrições das ações realizadas neles, enquanto as transições são indicadas por setas apontando a direção do fluxo. O estado inicial é representado por um círculo preenchido, enquanto o estado final, quando explicitado, é representado por um círculo preenchido circundado por um anel. Diagramas de estado constituem representações em alto nível do comportamento de um sistema ou de um de seus objetos frente a eventos internos ou externos (SOMMERVILLE, 2011).

2.5.10 Testes de Software

Os testes de *software* constituem uma atividade central da Engenharia de *Software*, responsável por verificar se um sistema atende aos requisitos especificados e por identificar defeitos antes que impactem o usuário final (SOMMERVILLE, 2011). Entre eles destacam-se:

- Testes unitários: verificam isoladamente o comportamento de uma função ou componente específico (PRESSMAN; MAXIM, 2016);
- Testes de integração: verificam a interação entre múltiplos componentes do sistema, incluindo rotas de API, banco de dados e RNs (PRESSMAN; MAXIM, 2016);
- Testes de contrato: verificam se a comunicação entre dois serviços distintos respeita o formato de dados e o comportamento esperado por ambas as partes, permitindo testar a integração entre serviços de forma isolada, sem a necessidade de implantá-los simultaneamente (LEHVÄ *et al.*, 2019).

Cada verificação realizada é formalizada por meio de Casos de Teste (CTs), que descrevem as condições, os dados de entrada e o resultado esperado, permitindo rastrear a cobertura da suíte de testes em relação aos RFs e às RNs do sistema. Em sistemas que lidam com dados sensíveis e múltiplos perfis de acesso, os testes também verificam segurança, confirmando que as regras de autorização e de isolamento de dados são respeitadas mesmo diante de tentativas de acesso indevido.

3 TRABALHOS RELACIONADOS

Este capítulo apresenta alguns trabalhos relacionados sobre modelos CADx voltados à classificação automática de imagens gastrointestinais e CDSSs com integração de modelos CAD, discutindo suas contribuições e lacunas de integração à rotina clínica.

3.1 Modelos CADx para Classificação de Imagens Gastrointestinais

Noor *et al.* (2023) desenvolveram um *framework* baseado em mecanismos de atenção, combinando as bases *Hyper-Kvasir* e *Kvasir V2*, totalizando 4.854 imagens distribuídas em cinco classes (Pólipo, Úlcera, Colite Ulcerativa, Esofagite e Normal), destacando as regiões acometidas nas imagens e reduzindo a dimensão das características extraídas por similaridade de cosseno, o que diminuiu o tempo de classificação e resultou em 98,07% de acurácia, 99,02% de precisão, 96,37% de sensibilidade e 97,68% de F1-score.

Já Mary *et al.* (2024) propuseram um modelo de classificação de anomalias gastrointestinais, utilizando a base *Kvasir V1* e sete categorias (não contemplando a Linha Z Normal), comparando uma CNN customizada com arquiteturas consolidadas (ResNet-50, VGG-19, Inception e Xception), alcançando 98,12% de acurácia, 95,23% de sensibilidade e 97,16% de especificidade. Contudo, trata-se de uma proposta restrita de classificação.

Do mesmo modo, Aslan (2025) propôs um modelo de *Ensemble Stacking*, combinando ResNet-50, DenseNet201 e MobileNetV3Large por meio de um meta-classificador, usando a base *Kvasir V2*, com 8.000 imagens distribuídas em oito classes. Esse modelo superou as CNNs individuais, alcançando 94,33% de acurácia, 94,36% de precisão, 94,27% de sensibilidade e 94,27% de F1-score, com maior dificuldade de diferenciação em classes de alta similaridade visual, como Esofagite Inicial e Linha Z Normal.

3.2 CDSSs com Integração de Modelos CAD

Um exemplo de integração na gastroenterologia é o sistema desenvolvido por Gong *et al.* (2023), que combina detecção (CAdE) e classificação (CADx) com um modelo de segmentação, integrados a um CDSS para uso em tempo real. O sistema alcançou 90,3% de acurácia em testes internos e 85,3% em testes com imagens de outra instituição. Em uma avaliação prospectiva com 522 exames de rotina, alcançou 89,3% de acurácia, sem diferença estatística significativa frente ao desempenho de dois endoscopistas experientes.

Considerando a reduzida integração voltada à gastroenterologia, também vale destacar iniciativas como a de Gonçalves *et al.* (2024), que apresentou o D.Iagnóstica, uma ferramenta CADx para diagnóstico de doenças pulmonares, composta por um modelo de classificação e uma aplicação *web* com cadastro de clínicas e profissionais de saúde. O modelo alcançou 97,27% de acurácia, 96,49% de precisão, 94,59% de sensibilidade e 95,51% de F1-*score*, reforçando a relevância de uma camada de aplicação que aproxime o modelo da rotina do usuário final.

Voltando ao contexto da gastroenterologia, Khryashchev (2025) apresentou a implementação de um CDSS para endoscopia baseado em YOLO e *Transformers*, testado em dois hospitais russos. O sistema oferece suporte tanto durante o exame, destacando regiões de interesse em tempo real, quanto para controle de qualidade pós-exame, por meio da anotação em vídeos. Em um dos centros avaliados, a metodologia de rotulagem semiautomática proposta reduziu de 35% a 45% o esforço de anotação de imagens em comparação à rotulagem manual.

Contudo, Ebad *et al.* (2025) apontam que a eficácia isolada desses modelos não é suficiente, sendo necessário avançar do campo teórico para aplicações reais, superando desafios de Engenharia de *Software*, como segurança e privacidade de dados, mau uso da IA e regulamentações de sistemas médicos. Confirmando essa lacuna, Nopour (2025) identificou que, de 217 estudos triados sobre CDSSs voltados ao câncer gástrico, apenas oito de fato se qualificavam para a revisão, sendo a maioria modelos isolados de avaliação, diagnóstico ou prognóstico, sem integração a uma plataforma de gestão clínica.

Apesar dessa lacuna de integração, a relevância clínica desses modelos permanece bem estabelecida: o ensaio clínico multicêntrico de Jeon *et al.* (2026), com 998 pacientes, demonstrou que a colonoscopia assistida por CADe eleva significativamente a taxa de detecção de adenomas (52,3% contra 36,1% no grupo controle). Em concordância, a revisão sistemática e meta-análise em rede de Shinozaki *et al.* (2026), reunindo 48 ensaios clínicos randomizados e quase 39 mil pacientes, também confirmou aumentos consistentes em relação à colonoscopia convencional, produzidas por múltiplos sistemas de CADe.

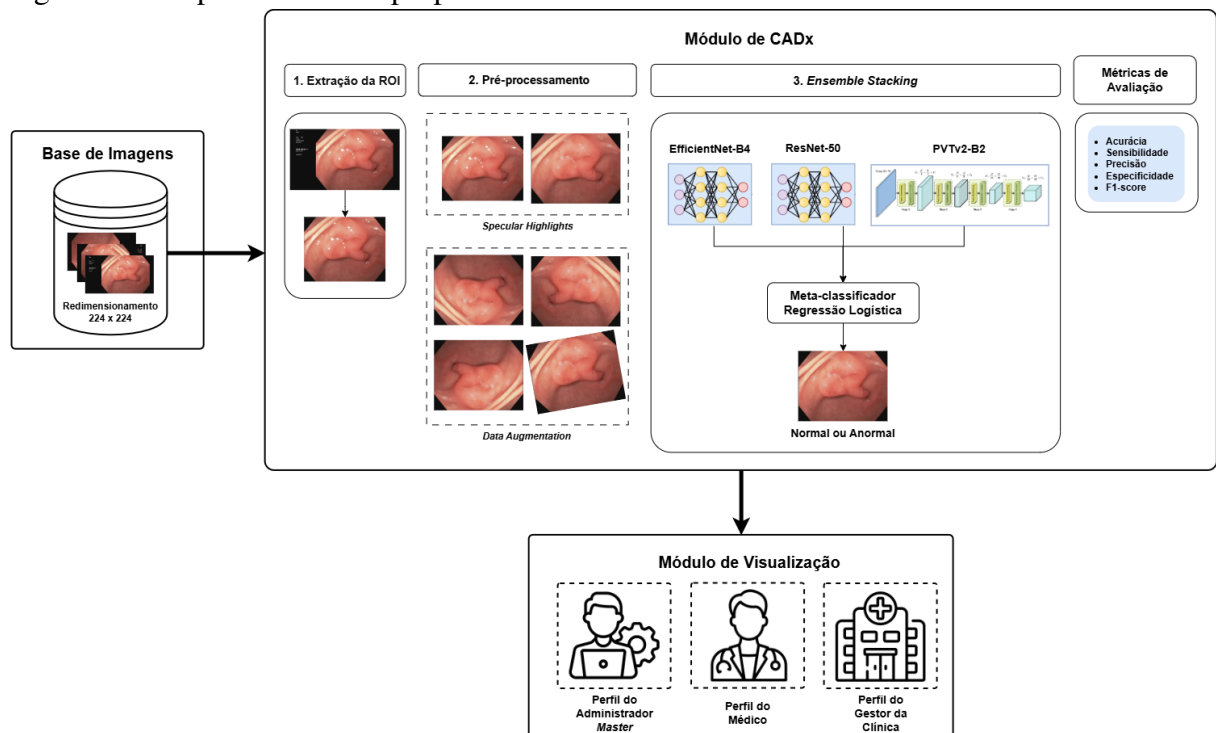
Os trabalhos apresentados evidenciam que, embora existam modelos CADx consolidados e sistemas já testados em ambiente clínico real (Gong *et al.* (2023) e Khryashchev (2025)), essas soluções ainda são escassas, sobretudo no contexto da gastroenterologia. Poucos trabalhos avançam para uma plataforma *web* completa que integre atividades organizacionais e médicas. Diante dessa lacuna, este trabalho propõe o ClinicAI, um protótipo de sistema *web* que busca aproximar os avanços algorítmicos da rotina prática das clínicas e dos profissionais de saúde.

4 MATERIAIS E MÉTODO PROPOSTO

O método proposto tem natureza aplicada, voltado ao desenvolvimento de um protótipo de solução *web* de suporte à gestão clínica e à tomada de decisão por meio da análise de exames endoscópicos por IA, denominado ClinicAI. Por se tratar de uma prova de conceito ainda não validada em ambiente clínico real, o desenvolvimento seguiu uma abordagem iterativa e incremental (SOMMERVILLE, 2011), com início no levantamento de requisitos do sistema, responsável por definir a estrutura arquitetural do *software* e orientar o desenvolvimento das etapas do método proposto.

A arquitetura do método proposto é composta por um módulo de CADx e um módulo de visualização, ilustrados na Figura 4. O módulo de CADx corresponde a reconstrução e treinamento de modelos com base no método proposto por Viana (2026), com o acréscimo do Grad-CAM. Já o módulo de visualização se refere à base do sistema de gestão clínica, com o desenvolvimento do *backend*, *frontend* e a modelagem de banco de dados, com acesso por meio de três perfis de usuário do sistema, *Administrator Master* (ADM), Médico (MED) e Gestor da Clínica (GDC), responsável por apresentar os resultados gerados pelo módulo de IA.

Figura 4 – Etapas do método proposto.



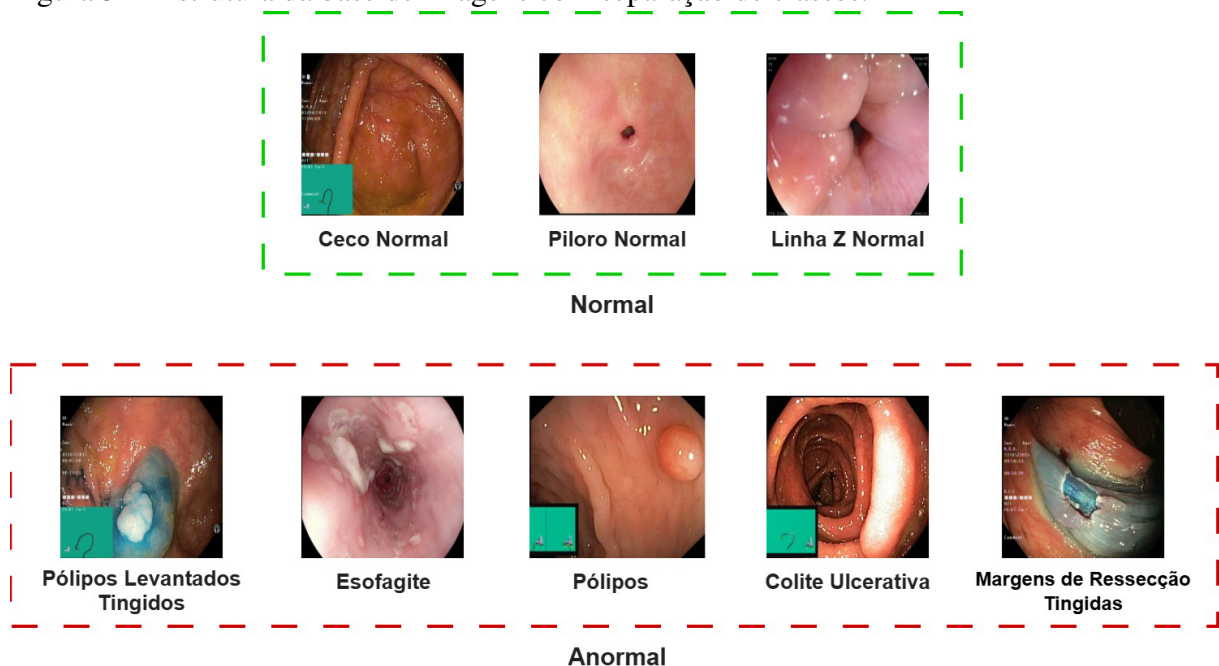
Fonte: Adaptado de Viana (2026).

4.1 Base de Imagens

Foi utilizada a base de imagens pública *Kvasir VI* no *Kaggle* (POGORELOV *et al.*, 2017), reconhecida na literatura como um dos principais conjuntos de dados abertos voltados à pesquisa em gastroenterologia e desenvolvimento de sistemas CAD. Esta base é composta por 4.000 imagens endoscópicas separadas em oito classes: Pólipos Levantados Tingidos (*Dyed Lifted Polyps*), Margens de Ressecção Tingidas (*Dyed Resection Margins*), Esofagite (*Esophagitis*), Ceco Normal (*Normal Cecum*), Píloro Normal (*Normal Pylorus*), Linha Z Normal (*Normal Z Line*), Pólipos (*Polyps*) e Colite Ulcerativa (*Ulcerative Colitis*), com 500 imagens cada, anotadas por especialistas médicos e totalizando 1.500 imagens normais e 2.500 anormais. As imagens possuem dimensões de 720x576 a 1920x1072 *pixels*, todas em formato JPG.

As imagens foram redimensionadas para 224x224 *pixels*. O método proposto utiliza uma abordagem de *Ensemble Stacking*, que requer o treinamento e a inferência de múltiplos modelos de alta complexidade. Essa prática reduz o custo computacional e o consumo de memória de vídeo (GPU), o que seria inviável com as resoluções originais diante de recursos computacionais limitados. Em paralelo, as características morfológicas fundamentais continuam preservadas, garantindo a viabilidade do método. A Figura 5 apresenta a estrutura da base de imagens com a separação das classes e exemplos de cada uma delas.

Figura 5 – Estrutura da base de imagens com separação de classes.



Fonte: Elaborado por Viana (2026).

4.2 Módulo de CADx

O módulo de CADx foi desenvolvido para auxiliar na triagem automática de exames endoscópicos classificando as imagens como normais ou anormais, combinando diferentes arquiteturas de DL que se complementam para aumentar a robustez e a precisão da classificação, sendo adaptado para atuar como um serviço integrado à arquitetura do sistema.

4.2.1 Extração de ROI

As imagens passam primeiro por uma etapa de preparação, descrita na Seção 4.1, e as que possuem áreas irrelevantes, como bordas pretas para anotações médicas, passam ainda pela extração da *Region of Interest* (ROI). Esse processo visa remover das imagens elementos que podem prejudicar o aprendizado e enviesar o modelo, identificando o maior retângulo possível que não contenha *pixels* pretos em suas laterais, mantendo a área útil da imagem.

4.2.2 Pré-Processamento

No pré-processamento, são realizadas duas etapas fundamentais. A primeira é a redução de *Specular Highlights* presentes na superfície da mucosa, que busca minimizar pontos de luz gerados pela iluminação do equipamento durante o exame endoscópico; assim como a extração da ROI, essa etapa é aplicada previamente por meio de um *script* de pré-processamento independente do treinamento. A segunda é a aplicação de *Data Augmentation* durante o treinamento, ampliando artificialmente a diversidade da base de imagens de treinamento por meio de transformações geométricas a cada época (*epochs*), aumentando a capacidade de generalização dos modelos e reduzindo *overfitting*. Nessa mesma etapa, as imagens também são normalizadas com a média e o desvio padrão da base ImageNet, mantendo a compatibilidade com a distribuição de dados esperada pelos modelos pré-treinados por *Transfer Learning*.

4.2.3 Ensemble Stacking

A classificação de imagens endoscópicas em normais e anormais é realizada por meio de *Ensemble Stacking* combinando as características extraídas das CNNs EfficientNet-B4 e ResNet-50 e do *Vision Transformer* PVTv2-B2, arquiteturas selecionadas por apresentarem o melhor desempenho individual entre os modelos avaliados. As saídas dos três modelos, treinados individualmente, são concatenadas e utilizadas como entrada de um meta-classificador baseado

em RL, responsável por mapear essas predições para a predição final, permitindo discernir padrões de erros específicos e atribuir maior relevância à rede com maior competência para cada amostra da base de imagens.

4.2.4 Métricas de Avaliação

A eficácia do método proposto por Viana (2026) foi avaliada por meio das métricas Acurácia (ACC), Sensibilidade (SEN), Precisão (PRE), Especificidade (ESP) e F1-score. Os experimentos demonstraram que as etapas de extração da ROI e de pré-processamento contribuíram para o aumento do desempenho do modelo, com o *Ensemble Stacking* superando os modelos individuais e outras estratégias de combinação, alcançando $96,20\% \pm 0,87\%$ de acurácia e $96,21\% \pm 0,86\%$ de F1-score na validação cruzada com $k = 10$ folds. Foi realizada uma análise estatística por meio dos testes de Wilcoxon e *t*-pareado, confirmando diferenças estatisticamente significativas ($p < 0,05$) em favor da abordagem proposta.

Portanto, no presente trabalho, a reprodução do método manteve as mesmas métricas de avaliação e os mesmos hiperparâmetros disponibilizados por Viana (2026) em seu estudo. Também foram seguidas as etapas de extração da ROI e as de pré-processamento propostas. A diferença foi na validação cruzada estratificada, realizada com $k = 3$ folds, em vez dos $k = 10$ folds do trabalho original. Essa redução foi devido às limitações de tempo e recursos computacionais, considerando também os resultados positivos encontrados nos experimentos originais. Para a implantação do modelo no sistema, foram utilizados os pesos correspondentes ao *fold* que apresentou o melhor desempenho.

4.3 Módulo de Visualização

O módulo de visualização corresponde à interface de gestão clínica, responsável por apresentar aos usuários os resultados gerados pelo módulo de IA e por disponibilizar os recursos de gerenciamento de clínicas, pacientes e exames. O acesso segue a lógica de RBAC, sendo dividido em ADM, MED e GDC, cada um com acesso restrito às funcionalidades correspondentes à sua atuação, conforme as RNs estabelecidas e a segurança e proteção de dados.

4.3.1 Perfil do Administrador Master

O ADM representa o nível mais alto de privilégio administrativo do sistema, responsável pela governança da plataforma como um todo. Esse perfil não está vinculado a nenhuma clínica em particular, podendo gerenciar múltiplas unidades cadastradas simultaneamente e realiza as seguintes operações:

- Gerenciar usuários, clínicas e perfis de acesso;
- Consultar a listagem de exames em nível de metadados;
- Visualizar os *logs* de auditoria do sistema;
- Acompanhar o módulo de IA por meio de métricas disponibilizadas no *dashboard*.

Em contrapartida, não possui acesso às imagens submetidas dos exames, aos resultados da análise de IA ou à revisão dos exames, restritos ao médico responsável, materializando o princípio do menor privilégio (Subseção 2.4.1). Mesmo concentrando o maior número de permissões administrativas do sistema, o ADM não acumula acesso ao conteúdo clínico, já que essa não é uma informação necessária ao exercício de sua função.

4.3.2 Perfil do Médico

O MED é o perfil médico do sistema, sendo vinculado a uma única clínica e realizando o acompanhamento de seus pacientes e exames, sem acesso a outros dados clínicos, mesmo dentro da mesma clínica. É esse perfil que opera o ciclo de vida do exame descrito na Subseção 5.5, desde a submissão à análise por IA até a conclusão da revisão clínica, realizando as operações de:

- Acompanhar as informações de seus pacientes e o histórico de atendimento;
- Acompanhar, por meio do *dashboard* do sistema, indicadores de volume e desempenho operacional referentes aos pacientes e exames sob sua responsabilidade;
- Visualizar, gerenciar e baixar os arquivos de exame e os resultados gerados pela IA;
- Enviar a imagem de um exame para análise automatizada pelo módulo de CADx, obtendo a classificação e o mapa de ativação Grad-CAM correspondente;
- Revisar e concluir o exame, registrando observações e apontamentos sobre a análise;
- Imprimir o relatório do exame já concluído em *Portable Document Format* (PDF).

Ao realizar o *upload* da imagem de um exame, o MED pode submetê-lo à análise automatizada por meio do módulo de CADx, definindo a normalidade ou não dos resultados

e destacando visualmente as regiões relevantes por meio de mapas de ativação Grad-CAM. O resultado pode ser visualizado, contendo as métricas encontradas e a imagem processada. O resultado gerado pela IA serve apenas de suporte, sendo a revisão do médico, e não o modelo, que determina o desfecho diagnóstico do exame, por meio de suas observações e apontamentos ao finalizar o fluxo (Subseção 5.5).

A visualização, o gerenciamento e o *download* dos arquivos de exame, assim como os resultados diagnósticos gerados pela IA, são reservados exclusivamente ao médico responsável, em conformidade com o Art. 73 da Resolução CFM nº 2.217/2018 (Conselho Federal de Medicina, 2018), que resguarda o sigilo profissional e veda ao médico revelar informações obtidas no exercício de sua profissão sem consentimento do paciente. A única exceção é a impressão do exame já concluído em PDF, também disponível ao GDC.

4.3.3 Perfil do Gestor da Clínica

O GDC é o perfil administrativo clínico, vinculado a uma única clínica, sem acesso aos dados de nenhuma outra, refletindo a arquitetura *multi-tenant* do sistema. Ele complementa o ADM na gestão do sistema, porém com escopo limitado e também sem acesso ao conteúdo clínico dos exames, realizando as operações de:

- Cadastrar e acompanhar médicos e pacientes vinculados à clínica;
- Acompanhar, por meio do *dashboard* do sistema, indicadores agregados de volume e desempenho operacional referentes a todos os pacientes e exames da clínica que gerencia;
- Imprimir o relatório do exame já concluído em PDF, contendo o resultado diagnóstico registrado pelo médico responsável.

A exceção da impressão do relatório do exame não representa acesso à interpretação ou à revisão do conteúdo do exame, que permanecem de responsabilidade exclusiva do médico. O GDC apenas viabiliza a entrega física de um documento cuja conclusão diagnóstica já foi integralmente registrada pelo MED, prática amparada pelo Art. 78 da Resolução CFM nº 2.217/2018 (Conselho Federal de Medicina, 2018), que reconhece a participação de auxiliares no manuseio de informações, condicionada à orientação e à responsabilidade do médico quanto à manutenção do sigilo profissional.

5 DESENVOLVIMENTO DO SISTEMA

Este capítulo detalha o processo que fundamentou a construção do método proposto, apresentado no Capítulo 4. O desenvolvimento foi conduzido por meio de atividades de Engenharia de *Software*, contemplando o levantamento de requisitos, a modelagem dos casos de uso, a definição da arquitetura do sistema, a modelagem do banco de dados e a especificação do fluxo de processamento dos exames.

5.1 Levantamento de Requisitos

O levantamento de requisitos do sistema foi conduzido por meio de uma combinação de *brainstorming*, análise documental e prototipação. O *brainstorming* permitiu identificar as necessidades de organização clínica, gerenciamento de usuários e integração com os recursos de IA aplicados à análise de exames endoscópicos. A análise documental teve como base o estudo do modelo de classificação proposto por Viana (2026), cuja adaptação ao contexto de um sistema de gestão clínica orientou a definição dos requisitos relacionados à análise automatizada. Por fim, a prototipação permitiu validar e refinar os requisitos identificados à medida que os módulos de classificação e visualização eram construídos.

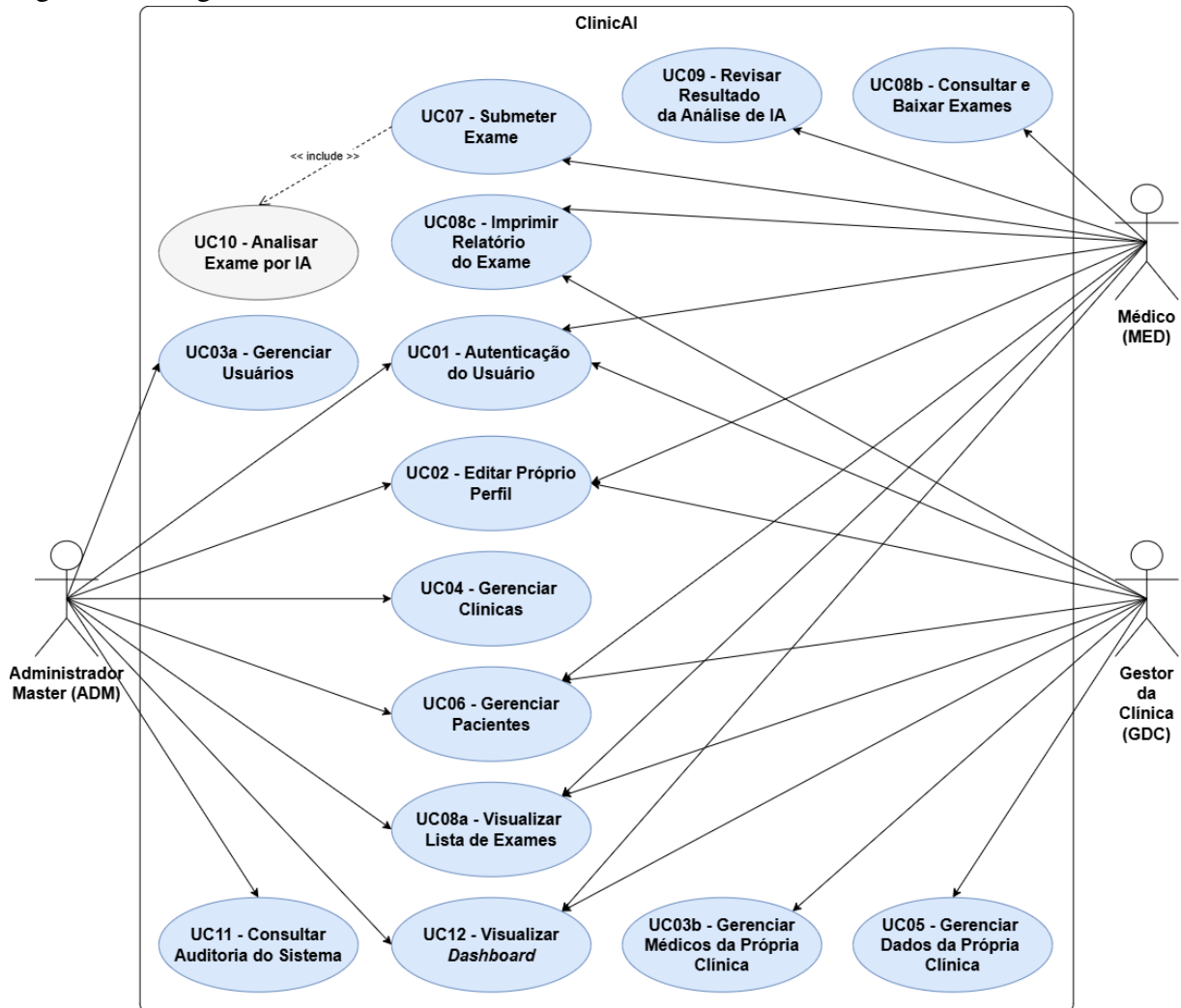
A partir desse processo, foram definidos os RFs (Apêndice A), os RNFs (Apêndice B) e as RNs (Apêndice C) que orientaram o desenvolvimento do módulo de visualização. Ao todo, os RFs foram organizados em oito módulos, sendo eles autenticação e controle de acesso, gerenciamento de usuários, de clínicas, de pacientes e de exames, análise de exames com IA, auditoria e rastreabilidade, e *dashboard* e monitoramento. Já os RNFs estão distribuídos entre segurança, persistência de dados, modularização, desempenho, escalabilidade, portabilidade, manutenibilidade e usabilidade, e as RNs em quatro categorias: cadastro e vínculos, fluxo de processamento de exames, análise de IA, e segurança, auditoria e controle de acesso.

5.2 Modelagem de Casos de Uso

A partir dos RFs levantados e dos perfis de usuário do sistema, também foram definidos os UCs e seus atores, detalhados no Apêndice D. Ao todo, foram definidos três atores, um correspondente a cada perfil de usuário, e 12 UCs. Destes, o UC03 e o UC08 foram subdivididos e diferenciados por letras, já que cada uma corresponde a uma mesma intenção geral, mas com escopo de atuação distinto. O UC03a (Gerenciar usuários) é de responsabilidade

do ADM sobre todos os usuários do sistema, enquanto o UC03b (Gerenciar médicos da própria clínica) é de responsabilidade do GDC, ao gerenciar os médicos vinculados a sua clínica. De forma semelhante, o UC08 se divide em visualizar a lista de exames (UC08a, comum aos três perfis), consultar e baixar exames (UC08b, exclusivo do MED) e imprimir o relatório do exame (UC08c, do MED e do GDC). A Figura 6 apresenta o diagrama de casos de uso elaborado.

Figura 6 – Diagrama de casos de uso.



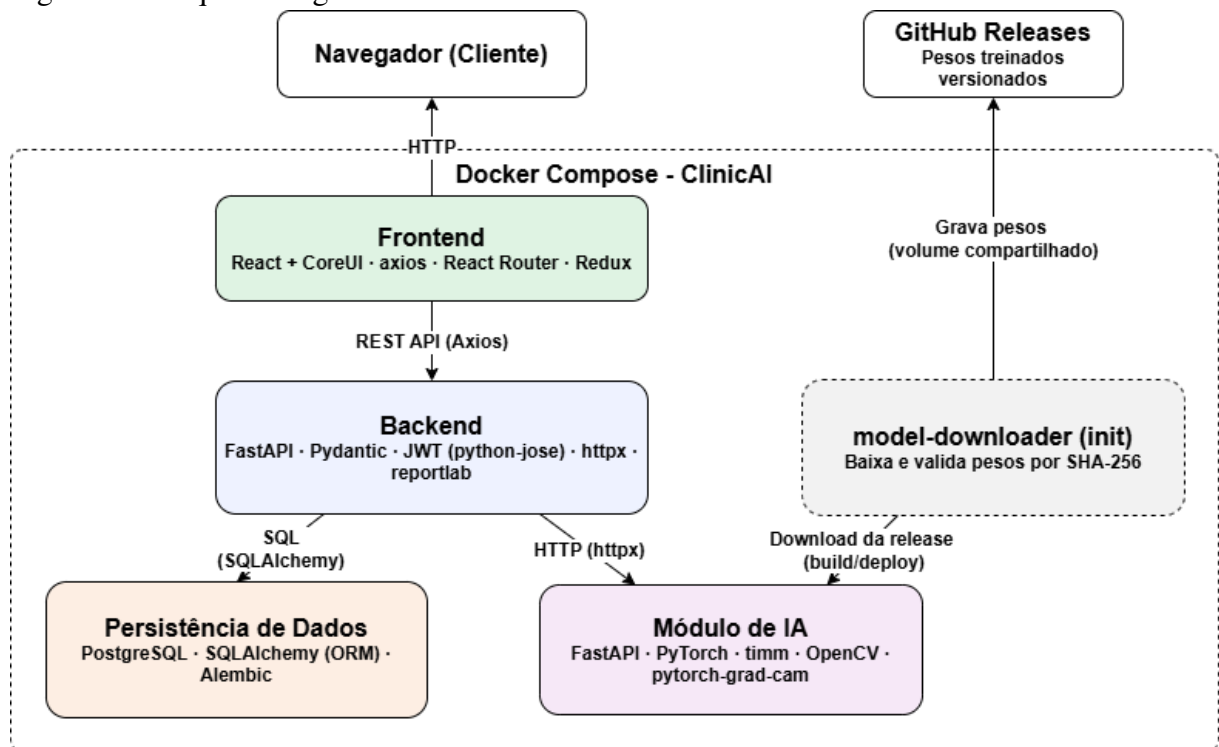
Fonte: Elaborado pela autora.

Alguns UCs merecem destaque por integrarem partes centrais de funcionamento do sistema. O UC07 (Submeter exame), ao ser executado pelo MED, desencadeia o UC10 (Analisar exame por IA) por meio da relação de *include*, delegando ao módulo de IA o processamento da imagem e a geração do resultado. Como o módulo de IA integra a arquitetura do sistema como um serviço interno, esse caso de uso não possui associação direta a um ator humano. Já o UC09 (Revisar resultado da análise de IA) é de responsabilidade exclusiva do MED, refletindo a regra de negócio de que apenas o perfil médico pode registrar conclusões clínicas dos exames.

5.3 Arquitetura do Sistema

A arquitetura do sistema foi projetada seguindo o modelo Cliente-Servidor e a arquitetura em camadas apresentados na Seção 2.5, sendo estruturada em quatro camadas modulares: *frontend*, *backend*, persistência e módulo de IA, cada uma isolada em seu próprio *container* Docker. Essa organização separa as responsabilidades entre os componentes, facilitando a manutenção e a escalabilidade do sistema, além de possibilitar a integração entre a interface *web* e os recursos computacionais responsáveis pela análise de imagens médicas, inicialmente direcionados ao processamento de exames endoscópicos. A Figura 7 apresenta a visão geral da arquitetura proposta para o desenvolvimento do ClinicAI, destacando seus principais componentes, as tecnologias utilizadas e o fluxo de comunicação entre eles. Cada uma dessas camadas, a comunicação entre elas e a infraestrutura de implantação são detalhadas nas subseções seguintes.

Figura 7 – Arquitetura geral do sistema.



Fonte: Elaborado pela autora.

5.3.1 Camada Frontend

O *frontend* é responsável pela interação do sistema com seus usuários, disponibilizando uma interface administrativa e clínica simples. Essa camada gerencia a apresentação das informações, permite o envio de requisições ao *backend* por meio do protocolo HTTP e exibe os

resultados retornados pela API, concentrando toda a lógica relacionada à experiência do usuário. Foi desenvolvido utilizando o React, uma biblioteca JavaScript voltada à construção de SPAs, escolhida por facilitar a manutenção e proporcionar maior fluidez na interação com o usuário.

Como *template*, foi utilizado o CoreUI React Admin, que oferece componentes visuais padronizados, como menus, tabelas, formulários e outros. As requisições HTTP ao *backend* são realizadas por meio da biblioteca Axios, a navegação entre telas é gerenciada pelo React Router, o estado global da aplicação é controlado com Redux, e os gráficos exibidos no *dashboard* são renderizados com Chart.js, integrado ao CoreUI.

5.3.2 Camada Backend

O *backend* concentra a lógica de negócio da aplicação, atuando como intermediário entre a interface *web*, o banco de dados e o módulo de IA. Entre suas responsabilidades estão a autenticação e autorização dos usuários por meio de *JSON Web Token* (JWT), validação dos dados recebidos, gerenciamento das requisições e coordenação do fluxo de processamento dos exames submetidos ao sistema. Ela também é responsável por encaminhar as imagens ao módulo de IA e disponibilizar os resultados processados para a interface da aplicação.

A implementação do *backend* foi através do FastAPI, um *framework* Python para desenvolvimento de APIs *web*. Ele foi escolhido por sua performance, simplicidade de uso, suporte à validação de dados e geração automática de documentação da API, além da versatilidade e adaptabilidade, permitindo futuras evoluções. A validação e a serialização dos dados são realizadas com Pydantic, nativamente integrado ao FastAPI, já a emissão e verificação dos *tokens* JWT utiliza a biblioteca Python-JOSE (*JS Object Signing and Encryption*), a comunicação com o módulo de IA é feita por meio da biblioteca HTTPX, e a geração do relatório de exame em PDF é feita por meio da biblioteca *reportlab*.

5.3.3 Persistência dos Dados

A persistência dos dados é realizada por meio de um banco de dados relacional, responsável pelo armazenamento das informações referentes aos usuários, clínicas, pacientes, exames e resultados das análises realizadas pelo sistema. Para sua execução foi escolhido o PostgreSQL, um SGBD relacional robusto e amplamente utilizado em aplicações *web*, por conta da sua confiabilidade, integridade e suporte a consultas estruturadas. O acesso aos dados é feito por meio do SQLAlchemy, utilizado como *Object-Relational Mapping* (ORM),

permitindo a manipulação das entidades da aplicação em Python e simplificando as chamadas de comandos *Structured Query Language* (SQL). O versionamento da estrutura do banco é gerenciado pelo Alembic, possibilitando o controle histórico das alterações de esquema ao longo do desenvolvimento por meio de *migrations*.

5.3.4 Módulo de IA

O módulo de IA corresponde à implementação, na arquitetura do sistema, do módulo de CADx apresentado na Seção 4.2. Reúne os componentes responsáveis pelo processamento e classificação das imagens médicas, sendo implementado como um serviço independente, exposto por meio de uma API própria e isolado em seu *container* dedicado. Sua estrutura contempla o *pipeline* de análise das imagens, incluindo etapas de pré-processamento, carregamento do modelo treinado, execução da inferência e geração dos resultados da classificação.

Para isso, foi desenvolvido utilizando bibliotecas especializadas: o PyTorch, empregado na construção, treinamento e execução dos modelos, devido à sua flexibilidade e ao amplo uso em pesquisas envolvendo redes neurais; o timm, utilizado na construção da arquitetura Vision Transformer (PVTv2-B2); o OpenCV, responsável pelas operações de processamento de imagens, como leitura, manipulação e preparação dos arquivos submetidos ao sistema; o Scikit-Learn, utilizado no carregamento e na execução do meta-classificador do *Ensemble Stacking* durante a inferência; e a biblioteca *pytorch-grad-cam*, utilizada na geração dos mapas de ativação Grad-CAM para explicação visual dos resultados.

5.3.5 Comunicação Entre os Componentes

A comunicação entre os diferentes componentes da aplicação, apresentados na Figura 7, ocorre por meio de APIs internas baseadas no protocolo HTTP. Quando um exame é submetido pelo usuário, a imagem é enviada do *frontend* ao *backend* por meio de uma requisição REST, utilizando a biblioteca Axios. O *backend* então realiza as validações necessárias e encaminha o arquivo para processamento pelo módulo de IA, por meio de uma requisição HTTP assíncrona, utilizando a biblioteca HTTPX. Após a conclusão da análise, os resultados obtidos são armazenados no banco de dados e disponibilizados novamente à interface da aplicação pela mesma rota REST inicial, permitindo consultas posteriores sem a necessidade de reproprocessamento da mesma imagem, o que contribui para a redução do tempo de resposta e do consumo de recursos computacionais. Já a geração do relatório do exame em PDF não envolve comunicação adicional

com o módulo de IA, sendo realizada somente pelo *backend* que reúne os dados do exame, da análise e as imagens já armazenadas para compor o documento diretamente, sem necessidade de novo processamento.

5.3.6 *Infraestrutura*

Com o objetivo de padronizar o ambiente de desenvolvimento, foi adotada a plataforma Docker para a organização da aplicação em *containers*, orquestrados por meio do Docker Compose. O *frontend*, *backend*, banco de dados e módulo de IA são organizados em *containers* independentes, possibilitando isolamento, maior portabilidade e facilidade de implantação. Essa abordagem também favorece a escalabilidade da arquitetura, permitindo que os componentes sejam atualizados, substituídos ou expandidos de maneira independente, reduzindo conflitos entre versões e facilitando futuras evoluções do sistema.

O módulo de IA pode ser executado tanto em *Central Processing Unit* (CPU) quanto em *Graphics Processing Unit* (GPU), porém, por padrão, os *containers* sobem em CPU. Além disso, como os pesos treinados não fazem parte do repositório devido ao seu volume, o *model-downloader* é responsável por baixá-los automaticamente de uma *release* do GitHub e validar sua integridade por meio de *checksums* SHA-256, disponibilizando-os ao módulo de IA.

5.4 Modelagem do Banco de Dados

O modelo de dados do sistema foi estruturado de forma relacional, refletindo as dez entidades identificadas durante o levantamento de requisitos (Apêndice A). A Tabela 1 descreve as entidades do modelo, suas tabelas correspondentes e seus papéis dentro do sistema.

Tabela 1 – Entidades do modelo de dados do sistema.

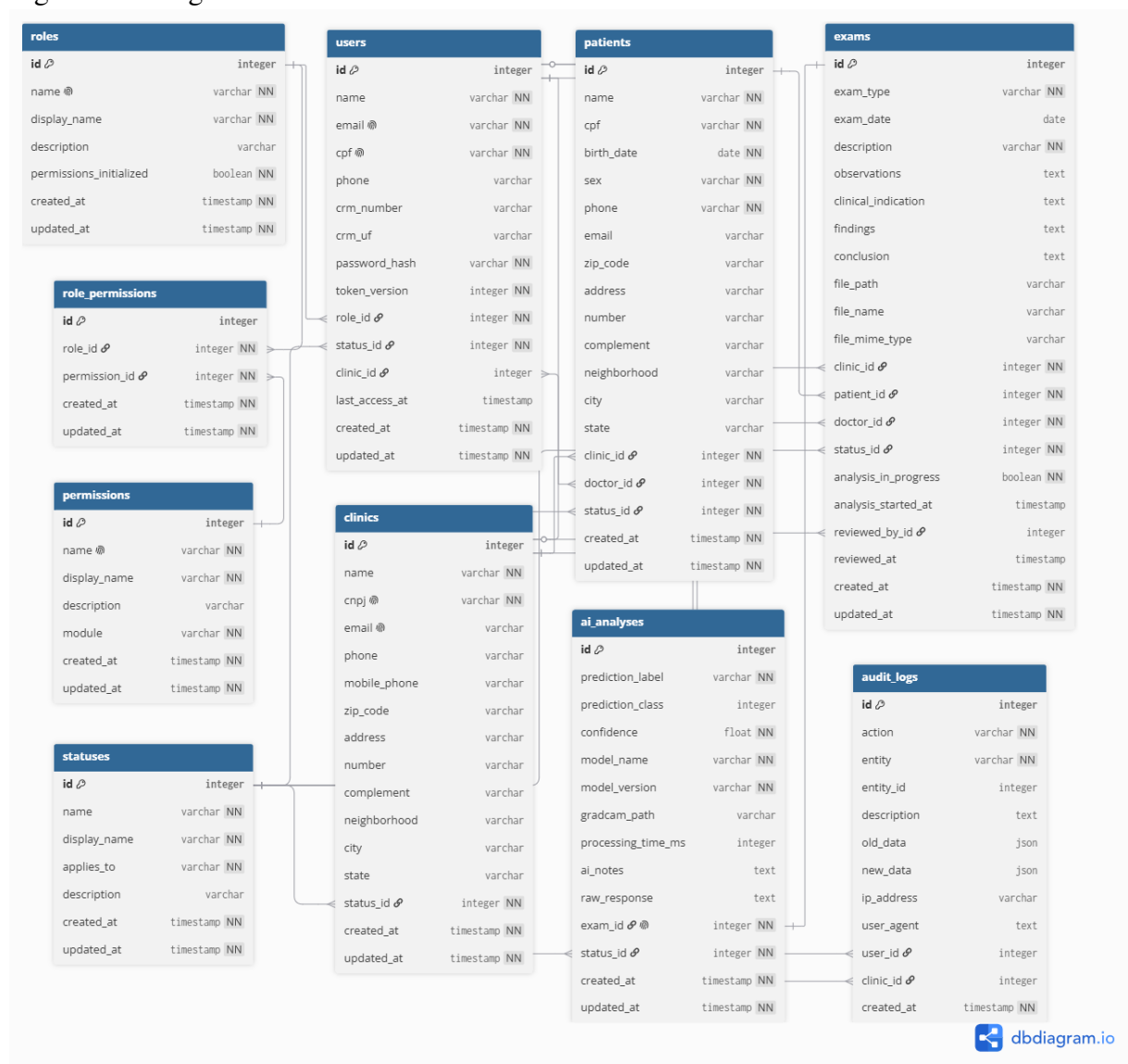
Entidade	Tabela	Descrição
Usuário	<i>users</i>	Armazena dados de autenticação e perfil de cada usuário, vinculado a um papel de acesso e, quando aplicável, a uma clínica.
Papel	<i>roles</i>	Define os perfis de acesso do sistema (ADM, MED, GDC).
Permissão	<i>permissions</i>	Lista as ações atribuídas a cada papel (ex.: visualizar clínicas, atualizar exames).
Papel-Permissão	<i>role_permissions</i>	Tabela associativa que implementa a relação N:N entre papéis e permissões, viabilizando o RBAC.
Status	<i>statuses</i>	Centraliza os possíveis estados de diferentes entidades (usuários, clínicas, pacientes, exames e análises de IA), diferenciados pelo campo <i>applies_to</i> .
Clínica	<i>clinics</i>	Armazena os dados cadastrais e de endereço das clínicas.
Paciente	<i>patients</i>	Armazena os dados dos pacientes, vinculados a uma clínica e a um médico responsável.
Exame	<i>exams</i>	Armazena os metadados e o arquivo de cada exame submetido, vinculado ao paciente, à clínica e ao médico responsável.
Análise de IA	<i>ai_analyses</i>	Armazena o resultado gerado pelo módulo de IA para cada exame (classe predita, confiança, mapa de calor Grad-CAM), em relação 1:1 com o exame.
Log de Auditoria	<i>audit_logs</i>	Registra as operações críticas do sistema, associando-as, quando aplicável, a um usuário e a uma clínica.

Fonte: Elaborado pela autora.

Entre as decisões de modelagem adotadas, destaca-se o uso de RBAC, implementado por meio de três entidades relacionadas (*roles*, *permissions* e *role_permissions*), permitindo associar múltiplas permissões a um mesmo papel de forma flexível, em vez de vincular permissões diretamente aos usuários. Além dessa, os estados possíveis das entidades do sistema foram centralizados em uma única entidade genérica de *status*, evitando a duplicação de tabelas e favorecendo a manutenibilidade do *schema*, diferenciando os *status* pela sua aplicação (*applies_to*), através de um campo na tabela específico para esse registro. Outro ponto é que cada exame possui uma relação de um para um (1:1) com sua respectiva análise de IA, garantindo que cada resultado gerado pelo módulo de IA permaneça diretamente vinculado ao exame que o originou.

As relações entre entidades seguem, em sua maioria, a cardinalidade um-para-muitos (1:N): uma clínica pode ter muitos usuários, pacientes e exames; um médico pode ser responsável por muitos pacientes e exames; e um paciente pode possuir muitos exames. Duas relações são opcionais: o vínculo de um usuário a uma clínica, permitindo a existência de um ADM não vinculado a nenhuma clínica específica; e o usuário responsável por um *log* de auditoria, permitindo registrar eventos não autenticados (como tentativas de *login* inválidas) ou disparados automaticamente pelo próprio sistema, sem comprometer a obrigatoriedade do registro em si. A Figura 8 apresenta o Diagrama de Entidade-Relacionamento do sistema.

Figura 8 – Diagrama de entidade-relacionamento do sistema



Fonte: Elaborado pela autora.

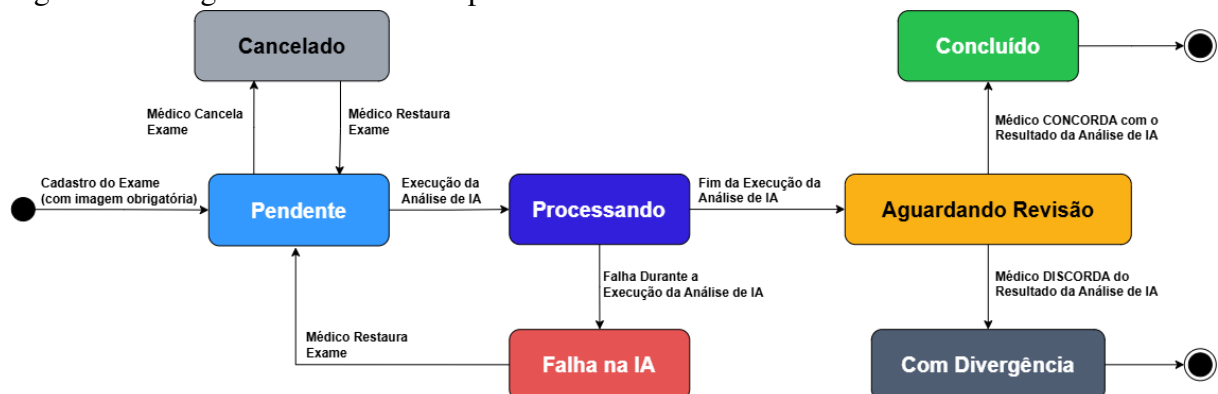
5.5 Fluxo de Processamento do Exame

O ciclo de vida de um exame no ClinicAI é representado por uma máquina de estados finita (Subseção 2.5.9), refletindo as transições entre os diferentes estágios do processamento, desde o cadastro até a conclusão da avaliação médica. O estado de cada exame é armazenado na entidade *status*, sendo atualizado automaticamente pelo sistema ou por meio de ações específicas realizadas pelos usuários autorizados (MED), sempre validadas pela máquina de estados antes da transição, conforme as RNs estabelecidas (Apêndice C).

O exame é cadastrado no *status* “Pendente”, já com a imagem obrigatoriamente anexada. Em seguida, o MED deve submeter explicitamente o exame à análise por IA, levando-o ao *status* “Processando”, enquanto o módulo de IA realiza a inferência sobre a imagem enviada. Concluída a inferência com sucesso, o *status* é atualizado automaticamente para “Aguardando Revisão”, ficando disponível para avaliação do médico responsável. Nessa etapa, ele analisa o resultado gerado, registra seu parecer clínico e conclui o exame, indicando se concorda ou não com a classificação automatizada, o que define o status final como “Concluído” (concorda) ou “Com Divergência” (discorda), encerrando o ciclo do exame.

Duas transições alternativas complementam o fluxo principal, ambas retornando o exame a “Pendente” para nova submissão. Em caso de falha durante a inferência, o exame passa ao *status* “Falha na IA”, podendo ser restaurado, retornando a “Pendente” para novo envio à análise. Da mesma forma, o exame pode ser cancelado quando está “Pendente”, sendo direcionado a “Cancelado”, e retornando a “Pendente” quando restaurado. A Figura 9 ilustra o diagrama de estados do exame no ClinicAI.

Figura 9 – Diagrama de estados do processamento do exame no sistema.



Fonte: Elaborado pela autora.

6 RESULTADOS E DISCUSSÃO

Neste capítulo são descritos os experimentos de validação do método proposto, com as métricas de desempenho obtidas e as interfaces da aplicação *web* desenvolvida. Além disso, são feitas comparações com os trabalhos relacionados.

6.1 Configuração Experimental

A aplicação *web* foi desenvolvida utilizando um notebook Dell G5 5590, com as seguintes especificações: processador Intel® Core™ i7-9750H de 9ª geração @ 2.60GHz, 16 GB de RAM, SSD de 480 GB, GPU NVIDIA GeForce GTX 1660 Ti 6 GB e sistema operacional Windows 11 Home Single Language 64 bits. O desenvolvimento em si utilizou técnicas e tecnologias descritas em mais detalhes na Seção 5.3, contando com o auxílio dos assistentes de codificação Codex e Claude Code na implementação, na depuração e teste do código-fonte do sistema. O projeto foi versionado em repositório no GitHub e os pesos treinados dos modelos por meio de GitHub Releases. Cada componente foi virtualizado por meio de *containers* Docker.

Já os modelos baseados em CNNs e ViTs foram treinados no ambiente virtual *Google Colab*, por ter sido o mesmo ambiente utilizado no código publicado por Viana (2026), por meio de *Jupyter Notebooks*, com a especificação de hardware GPU NVIDIA T4 Tensor Core, e conduzido em linguagem Python, utilizando principalmente as bibliotecas PyTorch e *timm* para a construção e o treinamento das arquiteturas, OpenCV para as etapas de pré-processamento das imagens, e Scikit-Learn para o treinamento do meta-classificador do *Ensemble Stacking* e o cálculo das métricas de avaliação.

Inicialmente os pesos foram pré-treinados na base ImageNet, empregando a técnica de *Transfer Learning*. O treinamento seguiu o protocolo de reprodução baseado no código publicado por Viana (2026), adotando os hiperparâmetros conforme descrito na Seção 4.2, sendo eles: otimizador *AdamW*, taxa de aprendizagem 0,0001, *batch size* de 64, função de perda *Categorical Crossentropy*, conduzido por até 300 épocas (*epochs*), com critério de *Early Stopping* de paciência 50 para evitar *overfitting*.

Para a reprodução do método *Ensemble Stacking* foi utilizada validação cruzada estratificada com $k = 3$ *folds*. A partir dos resultados de cada *fold*, foi realizado o cálculo da média e do desvio padrão entre eles. Os pesos integrados ao sistema vieram do *fold* que obteve o melhor desempenho entre os três.

6.2 Resultados da Reprodução do Módulo de CADx

A reprodução do método proposto por Viana (2026) permitiu identificar o *fold* de melhor desempenho, cujos pesos treinados foram integrados ao sistema, a partir da avaliação individual das três arquiteturas que o compõem, antes da combinação pelo meta-classificador, considerando os resultados obtidos nos $k = 3$ *folds*. A Tabela 2 resume o desempenho médio das três arquiteturas e do *Ensemble Stacking*.

Tabela 2 – Desempenho médio e desvio padrão dos modelos e do *Ensemble Stacking*.

Arquitetura	ACC (%)	PRE (%)	SEN (%)	ESP (%)	F1-score (%)
PVTv2-B2	95,58 ± 0,54	95,65 ± 0,47	95,58 ± 0,54	95,63 ± 0,37	95,59 ± 0,53
ResNet-50	94,90 ± 0,07	95,05 ± 0,13	94,90 ± 0,07	95,13 ± 0,24	94,93 ± 0,08
EfficientNet-B4	93,52 ± 1,31	93,56 ± 1,24	93,52 ± 1,31	93,21 ± 1,12	93,53 ± 1,29
Ensemble Stacking	95,97 ± 0,61	96,05 ± 0,51	95,97 ± 0,61	96,01 ± 0,31	95,99 ± 0,60

Fonte: Elaborado pela autora.

Assim como relatado por Viana (2026), a PVTv2-B2 foi a arquitetura individual de melhor desempenho médio nesta reprodução, superando as duas CNNs em todas as métricas avaliadas. A ResNet-50 apresentou o segundo melhor desempenho, com o menor desvio padrão entre as três arquiteturas, indicando o resultado mais estável entre os *folds*. Já a EfficientNet-B4 apresentou o menor desempenho médio entre as três e também o maior desvio padrão, indicando maior sensibilidade dessa arquitetura à divisão dos dados entre as execuções. O *Ensemble Stacking*, por sua vez, superou todas as arquiteturas individuais em todas as métricas, o que sugere um ganho de desempenho associado à combinação dos três modelos.

Para a integração ao sistema, foram utilizados os pesos treinados dos modelos no *fold* 1, cujos resultados indicam o melhor desempenho entre as três execuções, tanto em acurácia quanto em F1-score. A Tabela 3 apresenta o desempenho individual de cada arquitetura e do *Ensemble Stacking* nessa execução.

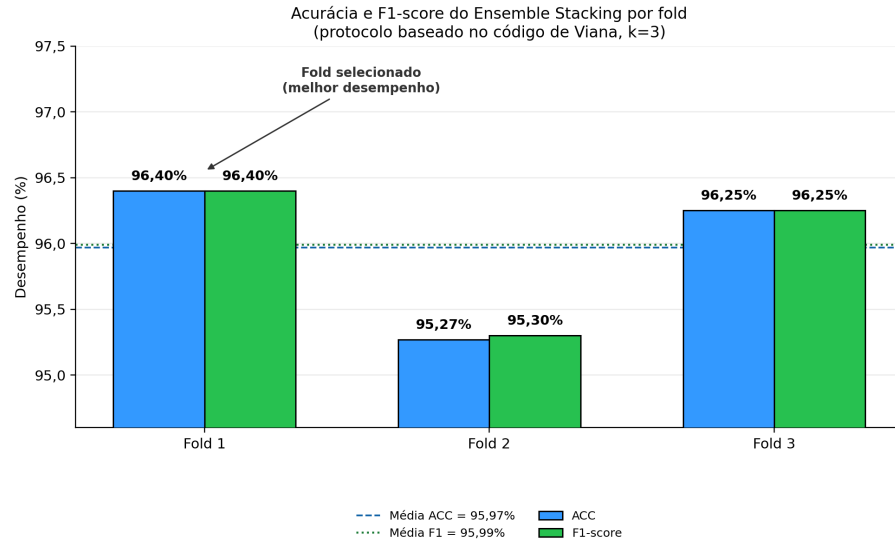
Tabela 3 – Desempenho do *fold* 1.

Modelo	ACC (%)	PRE (%)	SEN (%)	ESP (%)	F1-score (%)
PVTv2-B2	95,43	95,47	95,43	95,38	95,44
ResNet-50	94,83	94,90	94,83	94,86	94,85
EfficientNet-B4	94,53	94,53	94,53	94,14	94,53
Ensemble Stacking	96,40	96,41	96,40	96,24	96,40

Fonte: Elaborado pela autora.

A Figura 10 apresenta a acurácia e o F1-score do *Ensemble Stacking* obtidos em cada um dos três *folds*.

Figura 10 – Acurácia e F1-score do *Ensemble Stacking* por *fold*.

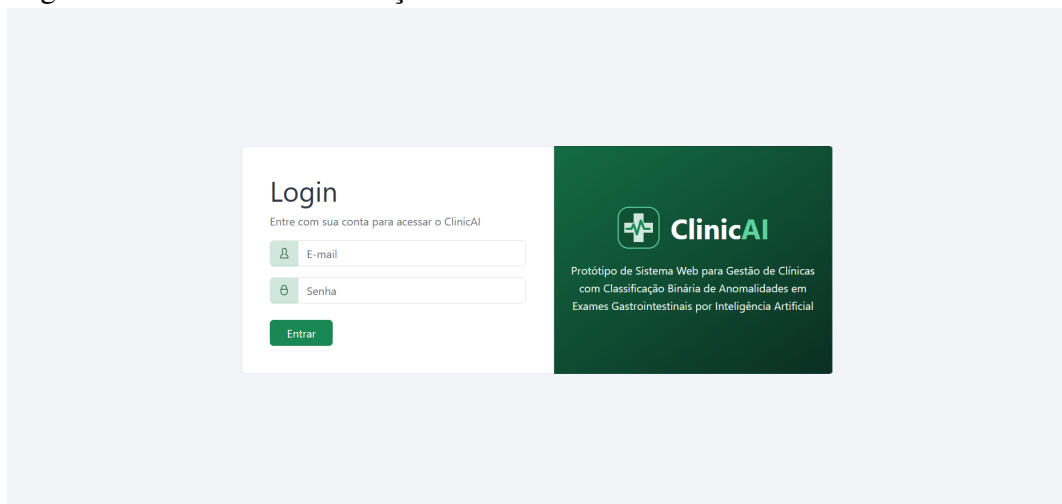


Fonte: Elaborado pela autora.

6.3 Resultados do Módulo de Visualização

As principais interfaces desenvolvidas para o sistema ilustram o fluxo de uso sob a perspectiva de cada perfil de usuário, bem como o ponto de integração entre o resultado da análise por IA e a decisão clínica do profissional médico. O acesso ao sistema é feito por meio de autenticação com *e-mail* e senha, conforme ilustrado na Figura 11.

Figura 11 – Tela de autenticação do sistema.

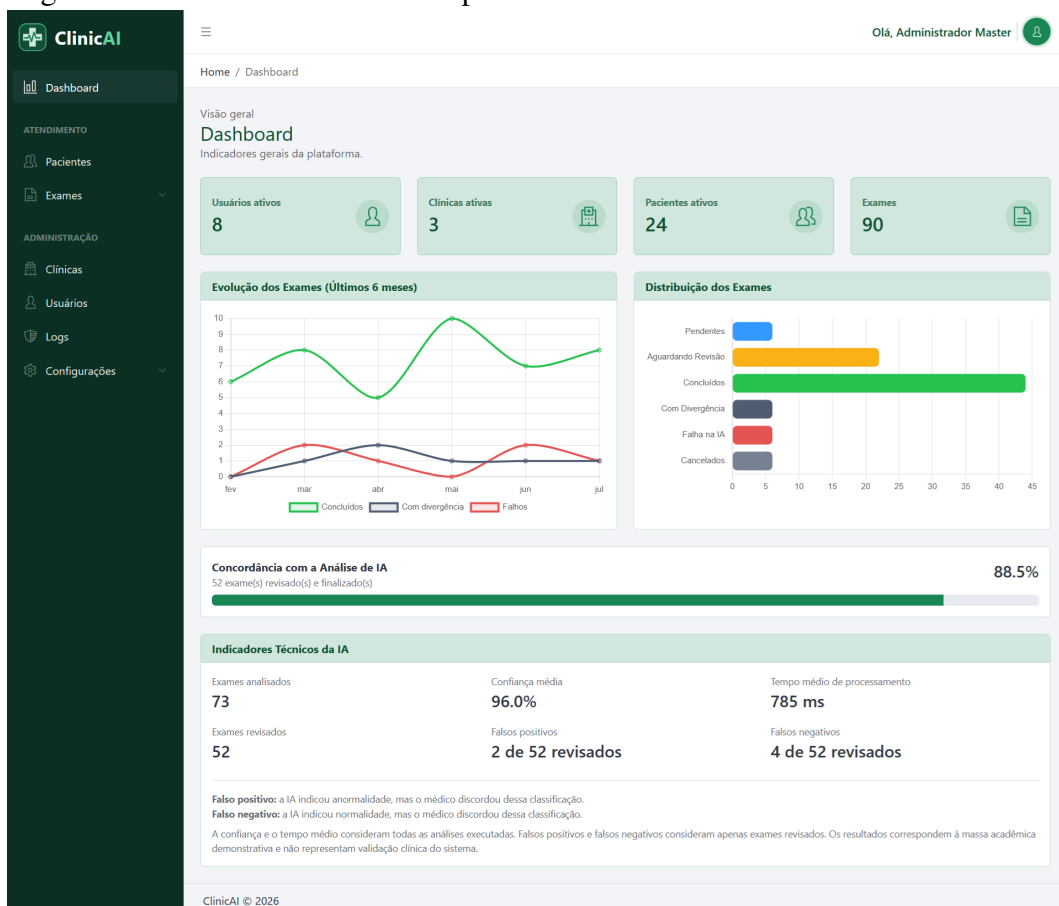


Fonte: Elaborado pela autora.

Após autenticado, o *dashboard* é exibido conforme o perfil do usuário logado. Para o ADM, ele reúne quatro indicadores gerais da plataforma (usuários, clínicas e pacientes ativos, e total de exames), um gráfico de “Evolução dos Exames” nos últimos seis meses por *status* (“Concluído”, “Com Divergência” e “Falha na IA”), a “Distribuição dos Exames” por *status*, e a “Concordância com a Análise de IA” (taxa calculada a partir dos exames concluídos em relação ao total de exames já revisados).

Exclusivamente para esse perfil, uma seção adicional de “Indicadores Técnicos da IA” apresenta o total de exames analisados, a confiança média das previsões, o tempo médio de processamento, o número de exames revisados, e a contagem de falsos positivos e falsos negativos, identificados a partir da divergência do médico em relação à classificação automatizada. A Figura 12 ilustra o *dashboard* do ADM.

Figura 12 – Tela de *dashboard* do perfil ADM.

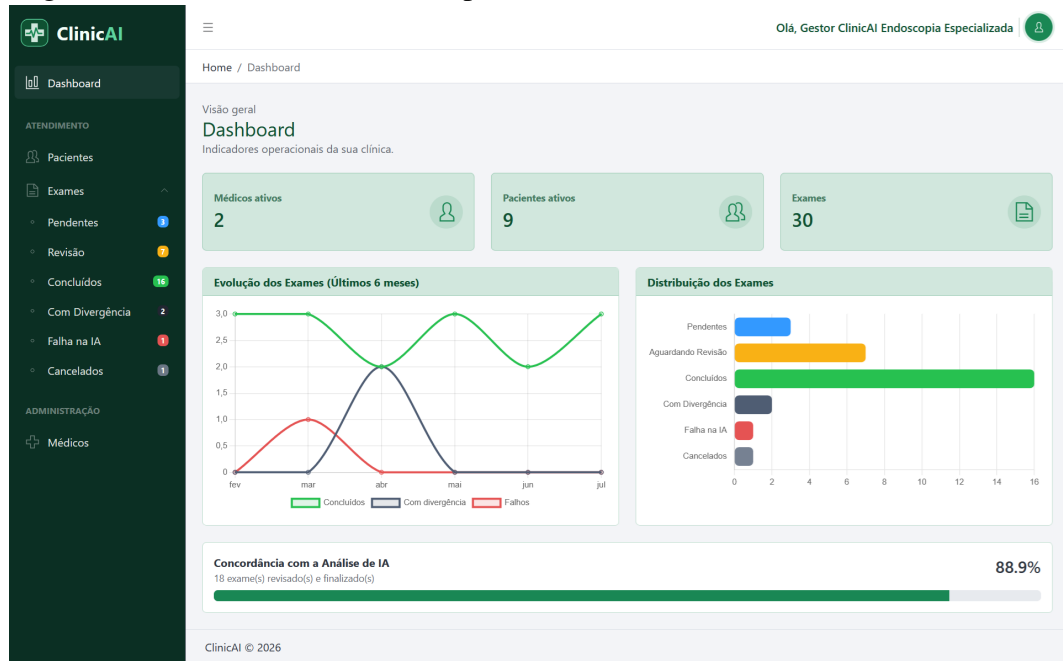


Fonte: Elaborado pela autora.

Os demais perfis têm acesso aos mesmos três gráficos (“Evolução dos Exames”, “Distribuição dos Exames” e “Concordância com a Análise de IA”), porém sem acesso à seção de “Indicadores Técnicos da IA”, exclusiva do ADM. O GDC visualiza os indicadores de

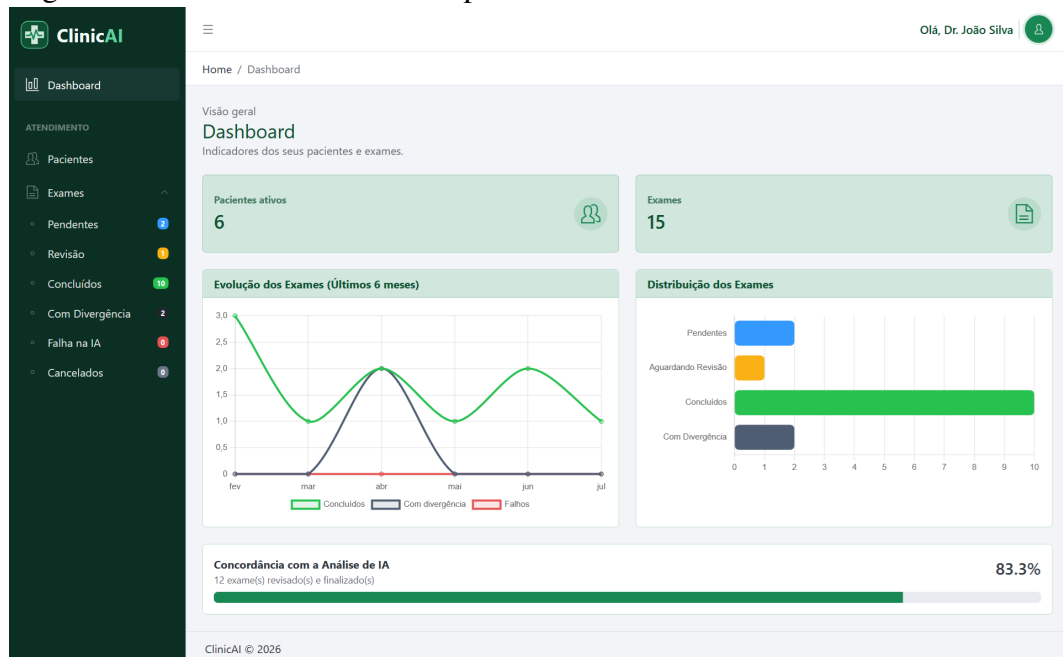
pacientes ativos e de exames restritos à clínica que gerencia, conforme a Figura 13, enquanto o MED visualiza os mesmos indicadores restritos aos seus próprios pacientes e exames sob sua responsabilidade (Figura 14).

Figura 13 – Tela de *dashboard* do perfil GDC.



Fonte: Elaborado pela autora.

Figura 14 – Tela de *dashboard* do perfil MED.



Fonte: Elaborado pela autora.

O perfil ADM concentra também o gerenciamento de papéis de usuário e permissões, na qual cada papel do sistema é associado ao conjunto de permissões que ele deve possuir, dessa forma concedendo ou removendo permissões de acesso a todos os usuários vinculados àquele papel, conforme a Figura 15.

Figura 15 – Tela de gerenciamento de papéis e permissões (RBAC).

ClinicAI

Olá, Administrador Master

Home / Perfis

Controle de Acesso

Editar Perfil

Configure os perfis oficiais e suas permissões de acesso.

Voltar

Esta tela permite alterar os dados e as permissões do perfil. Revise as informações antes de selecionar "Salvar".

Dados do Perfil

Perfil: clinic_manager

Nome de exibição: Gestor da Clínica

Nome técnico definido pelo catálogo oficial do sistema.

Descrição: Gestor responsável pela operação e pelos médicos da própria clínica.

Permissões do Perfil

Análises IA	Logs de Auditoria	Clinicas	Exames	Pacientes	Usuários
☐ Atualizar Análise por IA Permite atualizar ou revisar análise por IA.	☐ Visualizar Logs de Auditoria Permite visualizar registros de auditoria do sistema.	☐ Alterar Status das Clínicas Permite ativar, inativar ou bloquear clínicas.	☐ Alterar Status dos Exames Permite cancelar, concluir ou alterar status de exames.	☒ Alterar Status dos Pacientes Permite ativar, inativar ou bloquear pacientes.	☒ Alterar Status dos Usuários Permite ativar, inativar ou bloquear usuários.
☐ Criar Análise por IA Permite solicitar análise automatizada de exames por IA.		☐ Atualizar Clínicas Permite editar dados de clínicas.	☐ Atualizar Exames Permite editar dados de exames.	☒ Atualizar Pacientes Permite editar dados de pacientes.	☒ Atualizar Próprio Perfil Permite editar dados do próprio usuário autenticado.
☐ Visualizar Análise por IA Permite visualizar resultados de análises por IA.		☒ Atualizar Própria Clínica Permite editar os dados da clínica vinculada ao usuário autenticado.	☐ Baixar Exames Permite baixar os exames.	☒ Criar Pacientes Permite cadastrar novos pacientes.	☒ Atualizar Usuários Permite editar dados de usuários.
		☐ Criar Clínicas Permite cadastrar novas clínicas.	☒ Listar Exames Permite acompanhar a listagem e os status dos exames sem acessar detalhes clínicos.	☒ Visualizar Pacientes Permite visualizar pacientes cadastrados.	☒ Criar Usuários Permite cadastrar novos usuários no sistema.
		☐ Visualizar Clínicas Permite visualizar clínicas cadastradas.	☐ Revisar Exames Permite realizar a revisão médica do resultado da análise de IA.		☒ Visualizar Próprio Perfil Permite visualizar os dados do próprio usuário autenticado.
		☒ Visualizar Própria Clínica Permite visualizar os dados da clínica vinculada ao usuário autenticado.	☐ Upload de Exames Permite fazer o upload dos exames.		☒ Visualizar Usuários Permite visualizar usuários cadastrados.
			☐ Visualizar Exames Permite visualizar os detalhes clínicos dos exames cadastrados.		

Salvar Cancelar

ClinicAI © 2026

Fonte: Elaborado pela autora.

Com a definição e restrição dos perfis de acesso, o ADM pode realizar o gerenciamento de usuários, conforme ilustrado na Figura 16, na qual é possível cadastrar, editar, ativar e inativar, e associar cada usuário a um papel de acesso e, quando aplicável, a uma clínica.

Figura 16 – Tela de gerenciamento de usuários.

Fonte: Elaborado pela autora.

Além dessas atividades, o ADM é responsável pela consulta aos *logs* de auditoria do sistema, onde são registradas as operações críticas realizadas, informando o usuário responsável, a ação executada, a entidade afetada, a data e o horário do evento, e o endereço *Internet Protocol* (IP) da máquina utilizada, permitindo ao ADM reconstruir o histórico completo de eventos do sistema. A Figura 17 ilustra essa funcionalidade.

Figura 17 – Tela de *logs* de auditoria.

Fonte: Elaborado pela autora.

O GDC é responsável pela rotina operacional da unidade, cadastrando e acompanhando médicos e pacientes vinculados à sua clínica, sem acesso direto aos dados clínicos ou aos resultados dos exames pela interface do sistema, com exceção da impressão em PDF do relatório do exame concluído. A Figura 18 ilustra a tela de cadastro do médico pelo GDC.

Figura 18 – Tela de cadastro de médicos (perfil GDC).

Home / Usuários / Adicionar Usuário

Equipe Clínica

Cadastrar Médico

Gerencie os dados cadastrais do médico da sua clínica.

[Voltar](#)

Dados do Médico

Nome *

CRM *

UF do CRM *

Número do CRM

Selecione...

CPF *

Telefone

000.000.000-00

(88) 99999-9999

E-mail *

Senha *

Confirmar senha *

Permissões do usuário

O perfil Médico e a clínica são definidos automaticamente.

Fonte: Elaborado pela autora.

O MED também realiza o cadastro de pacientes, porém restrito aos pacientes sob sua responsabilidade clínica. A Figura 19 ilustra a tela de cadastro de pacientes do perfil MED.

Figura 19 – Tela de cadastro de pacientes (perfil MED).

Home / Pacientes / Adicionar Paciente

Registros de Saúde

Cadastrar Paciente

Cadastro clínico e administrativo do paciente.

[Voltar](#)

Dados do Paciente

Nome *

CPF *

000.000.000-00

Data de Nascimento *

Sexo *

Telefone/Celular *

E-mail

dd/mm/aaaa

Selecione

(88) 99999-9999

Clínica *

Médico Responsável *

ClinicAI Endoscopia Especializada

Dr. João Silva

CEP

Endereço

Número

00000-000

Complemento

UF

Fonte: Elaborado pela autora.

O ciclo de um exame começa a partir de um paciente pré-cadastrado e previamente selecionado ou não. O cadastro do exame é realizado exclusivamente pelo MED, que inclui os dados clínicos e administrativos do exame juntamente do *upload* da imagem endoscópica, ficando o exame no *status* “Pendente”. A Figura 20 ilustra a tela de cadastro de exame.

Figura 20 – Tela de cadastro de exame.

Fonte: Elaborado pela autora.

Após o cadastro, o MED é redirecionado a tela do exame, podendo então submeter o exame “Pendente” à análise por IA através do botão “Executar Análise de IA” e realizar a inferência. A Figura 21 ilustra a tela de exame “Pendente”.

Figura 21 – Tela de exame “Pendente”.

Fonte: Elaborado pela autora.

O resultado da inferência é apresentado mostrando a imagem original do exame com o mapa de ativação Grad-CAM ao lado, calculado a partir da ponderação dos mapas individuais

Uma vez finalizado, o exame pode ser gerado em PDF pelos perfis MED e GDC, reunindo os dados do paciente, do exame, o parecer médico e as imagens, conforme a Figura 23.

Figura 23 – Relatório do exame.

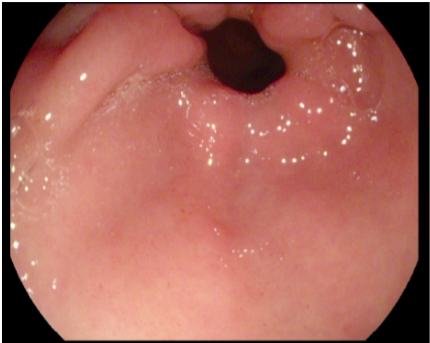
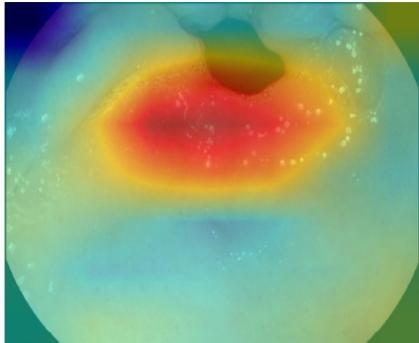

ClinicAI
 Sistema de Suporte à Análise de Exames

Relatório do Exame

Paciente e Exame		Data do Exame: 06/04/2026
Clínica ClinicAI Endoscopia Especializada Paciente Maria Oliveira Tipo de Exame Endoscopia digestiva alta	Médico Responsável Dr. João Silva CPF 700.000.101-11 Data de Nascimento 04/03/1983	

Revisão Médica	Data da Revisão: 06/04/2026
Achados Resultado da IA confirmado na revisão demonstrativa. Conclusão Revisão acadêmica concluída com concordância.	

Análise Automatizada		
Resultado Normal	Confiança 95%	Modelo ClinicAI ES Gastrointestinal

Imagens do Exame	
Imagem Original 	Mapa de Atribuição  Baixa Ativação Alta Ativação

Aviso: o resultado automatizado é um recurso de suporte e não substitui a avaliação do profissional responsável.

6.4 Comparação com a Literatura

A Tabela 4 resume o desempenho relatado pelos estudos apresentados no Capítulo 3, tomando como principal referência o trabalho de Viana (2026), do qual o método de classificação deste trabalho foi diretamente reproduzido, além de indicar quais deles avançam para além do modelo de classificação isolado e entregam uma aplicação ou sistema integrado.

Tabela 4 – Comparação do método proposto com trabalhos relacionados.

Trabalho	Base de Imagens	Método	ACC	F1-score	Sistema*
Gong <i>et al.</i> (2023)	Clínico (interno, 1.869 imagens, 6 classes)	EfficientNet-B3 (classificação de lesões)	90,3%	–	Sim
Noor <i>et al.</i> (2023)	<i>Hyper-Kvasir</i> e <i>Kvasir V2</i>	Atenção + redução por similaridade de cosseno	98,07%	97,68%	Não
Gonçalves <i>et al.</i> (2024)	Raios-X de tórax (Kaggle, 3 repositórios, 4 classes)	CNN customizada	97,27%	95,51%	Sim
Mary <i>et al.</i> (2024)	<i>Kvasir V1</i> (7 classes)	CNN customizada vs. ResNet-50/VGG-19/Inception/Xception	98,12%	–	Não
Aslan (2025)	<i>Kvasir V2</i> (8 classes)	<i>Ensemble Stacking</i> (DenseNet201, ResNet-50 e MobileNetV3Large)	94,33%	94,27%	Não
Khryashchev (2025)	Clínico (vídeos endoscópicos, dois hospitais russos)	YOLO + <i>Transformers</i> (detecção em tempo real e anotação semiautomática)	–	–	Sim
Método Proposto (ClinicAI)	<i>Kvasir V1</i>	<i>Ensemble Stacking</i> (reprodução Viana (2026), $k = 3$)	95,97%	95,99%	Sim

Fonte: Elaborado pela autora.

*Indica se o modelo de classificação foi integrado a uma aplicação ou sistema operacional, não implicando necessariamente RBAC ou fluxo de revisão médica, combinação presente no ClinicAI e não identificada nos demais trabalhos analisados.

Em relação à acurácia isolada do modelo de classificação, o ClinicAI ficou abaixo do resultado dos trabalhos focados exclusivamente na classificação (Mary *et al.* (2024) e Noor *et al.* (2023)), reflexo das diferenças metodológicas. Ainda assim, o desempenho permaneceu superior ao de Aslan (2025), que também emprega *Ensemble Stacking*, e comparável às faixas de acurácia relatadas por sistemas testados em ambiente clínico real, como Gong *et al.* (2023).

Entretanto, nenhum dos trabalhos com desempenho de classificação competitivo

(Aslan (2025), Mary *et al.* (2024) e Noor *et al.* (2023)) avança para uma aplicação ou sistema integrado. Gong *et al.* (2023) e Khryashchev (2025) são os dois exemplos voltados a endoscopia gastrointestinal identificados como sistemas testados em ambiente clínico real. O primeiro concentra-se num CDSS de apoio em tempo real durante o exame, enquanto o segundo atua para reduzir o esforço de anotação de imagens por meio de rotulagem semiautomática, sem reportar métricas de acurácia ou *F1-score* de classificação, já que seu foco está na eficiência do processo de anotação, não no desempenho de um modelo classificador. Nenhum dos dois, porém, inclui um módulo de gestão clínica associado.

Já Gonçalves *et al.* (2024), o trabalho mais próximo em estrutura por combinar um modelo CADx a uma aplicação *web* com cadastro de clínicas e profissionais, é voltado a doenças pulmonares, não à gastroenterologia. Portanto, o ClinicAI se diferencia por integrar o modelo de classificação a um sistema de gestão clínica completo, com RBAC, fluxo de revisão médica e explicabilidade via Grad-CAM, preenchendo a lacuna identificada nos trabalhos relacionados referente à escassez de plataformas *web* completas voltadas especificamente à gastroenterologia.

6.5 Testes Realizados

Para verificar o funcionamento correto dos principais fluxos do ClinicAI, foi desenvolvida uma suíte de testes automatizados com o *framework* Pytest, abrangendo testes unitários, de integração e de contrato (Subseção 2.5.10), num total de 33 CTs organizados por módulo funcional e relacionados aos requisitos e RNs (Apêndice E). Entre os pontos mais sensíveis cobertos estão o isolamento de dados entre clínicas, a exclusividade da revisão médica para o MED e a impressão do relatório do exame, a integridade e a privacidade dos registros de auditoria, e o comportamento do fluxo do exame sob condições de concorrência, incluindo a rejeição de transições não previstas entre *status* (Subseção 5.5).

7 CONCLUSÕES E TRABALHOS FUTUROS

A análise automática de imagens endoscópicas tem sido cada vez mais explorada como forma de apoiar profissionais de saúde na triagem de anormalidades gastrointestinais. Contudo, a maioria dos esforços concentra-se predominantemente em modelos de classificação isolados, com poucas iniciativas avançando para uma plataforma *web* completa que integre esses modelos ao fluxo organizacional e clínico. Diante dessa lacuna, este trabalho apresentou o ClinicAI, um protótipo de sistema *web* que integra um módulo de CADx, baseado na reprodução do método proposto por Viana (2026), a um sistema de gestão clínica.

O módulo de CADx, reproduzido a partir do código publicado pelo autor de referência, alcançou $95,97\% \pm 0,61\%$ de acurácia e $95,99\% \pm 0,60\%$ de *F1-score* com $k = 3 \text{ folds}$, resultado próximo ao obtido pelo autor em validação cruzada com $k = 10 \text{ folds}$, reforçando a validade da reprodução. Os pesos integrados ao sistema foram selecionados pelo *fold* cuja execução teve o melhor desempenho. Já o módulo de visualização integra esse modelo a um sistema de gestão clínica com três perfis de usuário e RBAC, no qual o resultado da análise automatizada é apresentado ao médico responsável como suporte à decisão, sem substituir o julgamento clínico, já que a conclusão do exame permanece de responsabilidade exclusiva do profissional médico.

Algumas limitações devem ser consideradas na interpretação destes resultados. O módulo de CADx foi validado exclusivamente sobre a base de imagens *Kvasir VI*, sem avaliação em bases de imagens externas, o que limita conclusões sobre sua capacidade de generalização para outros contextos de aquisição de imagens endoscópicas. Em segundo lugar, o ClinicAI é um protótipo ainda não validado em ambiente clínico real, não tendo sido submetido a testes com usuários finais nem a avaliações de usabilidade formais, o que é necessário antes de qualquer consideração sobre sua adoção prática.

Como trabalhos futuros, planeja-se:

- Explorar técnicas de realce das imagens na etapa de pré-processamento, complementando a extração da ROI e a redução de *Specular Highlights* já empregadas;
- Avaliar meta-classificadores alternativos à Regressão Logística no *Ensemble Stacking*;
- Validar o módulo de CADx em bases de imagens endoscópicas externas à *Kvasir VI*, avaliando sua capacidade de generalização;
- Explorar estratégias de otimização automatizada de hiperparâmetros e de balanceamento de classes;

- Implementar o envio automático dos exames para análise por IA no momento do cadastro, eliminando a necessidade de acionamento manual pelo MED enquanto o exame está no status “Pendente”, com o processamento organizado em uma fila que absorva picos de fluxo de submissões, mantendo o médico informado sobre a transição de “Pendente” para “Processando” e sendo notificado por alerta assim que o resultado estiver disponível;
- Implementar o processamento assíncrono das análises de IA, reduzindo o tempo de resposta percebido pelo usuário;
- Adicionar um módulo de acesso para o paciente, como um quarto perfil de usuário do sistema, permitindo a visualização dos próprios exames e resultados;
- Expandir o registro de modelos de IA do sistema para contemplar outros domínios clínicos além do gastrointestinal, aproveitando a arquitetura já adotada;
- Conduzir testes de usabilidade e validação clínica com médicos e equipes de clínicas parceiras, em uso real e prolongado do sistema;
- Realizar a implantação do sistema em um ambiente de produção real, seja em um servidor ou plataforma de nuvem tradicional, seja por meio de uma versão de demonstração pública em plataformas como o Hugging Face Spaces, que suporta a publicação de aplicações baseadas em *containers* Docker.

REFERÊNCIAS

- AKOLGO, E. A.; KORDA, D. R.; DAPAAH, E. O. Artificial intelligence in healthcare: A fusion of technologies. **Journal of Computer and Communications**, Scientific Research Publishing (SCIRP), v. 12, n. 12, p. 116–133, 2024. ISSN 2327-5219. Disponível em: <<https://doi.org/10.4236/jcc.2024.1212008>>.
- ALA'ANZY, M. A.; YESHPATOV, Z. A performance and scalability evaluation of monolithic (Django) vs. microservice (FastAPI) architectures for asynchronous API workloads in Python. In: **2026 18th International Conference on Electronics, Computer, and Computation (ICECCO)**. Kaskelen, Kazakhstan: [s.n.], 2026.
- Alembic. **Welcome to Alembic's documentation**. 2026. <<https://alembic.sqlalchemy.org/>>. Acesso em: 19 jul. 2026.
- ALEXIUK, M.; ELGUBTAN, H.; TANGRI, N. Clinical decision support tools in the electronic medical record. **Kidney International Reports**, Elsevier, v. 9, n. 1, p. 29–38, 2024. ISSN 2468-0249.
- ALI, T. E.; ALI, F. I.; HEKMAT, S.; EYVAZOV, F.; DAKIĆ, P.; ZOLTAN, A. D. A comparative evaluation of PostgreSQL, MongoDB, and MySQL for ethiopian migrant data management. In: **SQAMIA 2025: Workshop on Software Quality, Analysis, Monitoring, Improvement, and Applications**. Maribor, Slovenia: [s.n.], 2025. ISSN 1613-0073. CEUR Workshop Proceedings.
- AMBLER, S. W.; SADALAGE, P. J. **Refactoring Databases: Evolutionary Database Design**. Boston, MA: Addison-Wesley Professional, 2006. ISBN 0-321-29353-3.
- ARAÚJO, J. S.; HORNUNG, J. A. Inteligência artificial no diagnóstico de doenças: A responsabilidade civil do médico em caso de erro de diagnóstico. **Revista Themis**, v. 20, n. 1, p. 113–145, jan 2022. ISSN 1984-4867.
- ASLAN Şehmus. Ensemble-based deep transfer learning for robust gastrointestinal endoscopy image classification. **Balkan Journal of Electrical and Computer Engineering**, Balkan Journal of Electrical and Computer Engineering, v. 13, n. 1, p. 1–10, mar. 2025. ISSN 2147-284X. Disponível em: <<https://doi.org/10.17694/bajece.1630294>>.
- ASO, M. C.; SOSTRES, C.; LANAS, A. Artificial intelligence in GI endoscopy: what to expect. **Frontiers in Medicine**, Frontiers Media, v. 12, p. 1588873, 2025. ISSN 2296-858X.
- AWRAHMAN, B. J.; FATAH, C. A.; HAMAAMIN, M. Y. A review of the role and challenges of big data in healthcare informatics and analytics. **Computational Intelligence and Neuroscience**, Hindawi, v. 2022, p. 1–10, 2022.
- AYAN, E. Classification of gastrointestinal diseases in endoscopic images: Comparative analysis of convolutional neural networks and vision transformers. **Journal of the Institute of Science and Technology**, v. 14, n. 3, p. 988–999, 2024. ISSN 2146-0574.
- BEZEMER, C.-P.; ZAIDMAN, A. Multi-tenant saas applications: maintenance dream or nightmare? In: **Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE)**. Antwerp, Belgium: ACM, 2010. p. 88–92.

BORGLI, H.; THAMBAWITA, V.; SMEDSRUD, P. H.; HICKS, S.; JHA, D.; ESKELAND, S. L.; RANDEL, K. R.; POGORELOV, K.; LUX, M.; NGUYEN, D. T. D.; JOHANSEN, D.; GRIWODZ, C.; STENSLAND, H. K.; GARCIA-CEJA, E.; SCHMIDT, P. T.; HAMMER, H. L.; RIEGLER, M. A.; HALVORSEN, P.; LANGE, T. de. HyperKvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy. **Scientific Data**, Nature Publishing Group, v. 7, n. 1, p. 283, 2020. ISSN 2052-4463.

Brasil. **Lei nº 13.709, de 14 de agosto de 2018: Lei Geral de Proteção de Dados Pessoais (LGPD)**. Brasília: Diário Oficial da União, 2018. Disponível em: <https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm>.

BUTPHENG, C.; YEH, K.-H.; XIONG, H. Security and privacy in iot-cloud-based e-health systems—a comprehensive review. **Symmetry**, v. 12, n. 7, p. 1191, 2020.

BUTT, A. U. R.; MAHMOOD, T.; SABA, T.; BAHAI, S. A. O.; ALAMRI, F. S.; IQBAL, M. W.; KHAN, A. R. An optimized role-based access control using trust mechanism in e-health cloud environment. **IEEE Access**, IEEE, v. 11, p. 138813–138826, 2023.

CHACON, S.; STRAUB, B. **Pro Git**. 2. ed. Berkeley, CA: Apress, 2014. ISBN 978-1-4842-0077-3. Disponível em: <<https://git-scm.com/book>>.

CHADDAD, A.; PENG, J.; XU, J.; BOURIDANE, A. Survey of explainable ai techniques in healthcare. **Sensors**, MDPI, v. 23, n. 2, p. 634, 2023. ISSN 1424-8220. Disponível em: <<https://doi.org/10.3390/s23020634>>.

CHEN, Z.; LIANG, N.; ZHANG, H.; LI, H.; YANG, Y.; ZONG, X.; CHEN, Y.; WANG, Y.; SHI, N. Harnessing the power of clinical decision support systems: Challenges and opportunities. **Open Heart**, BMJ Publishing Group, v. 10, n. 2, p. e002432, 2023. Disponível em: <<https://openheart.bmj.com/content/10/2/e002432>>.

CHHETRI, S. K. **Comparative Study of Front-end Frameworks: React and Angular**. Dissertação (Bachelor's Thesis) — Oulu University of Applied Sciences, Oulu, Finland, 2024.

Conselho Federal de Medicina. **Resolução CFM nº 2.217, de 27 de setembro de 2018: Aprova o Código de Ética Médica**. 2018. <https://www.in.gov.br/materia/-/asset_publisher/Kujrw0TZC2Mb/content/id/48226289/do1-2018-11-01-resolucao-n-2-217-de-27-de-setembro-de-2018-48226042>. Diário Oficial da União, Brasília, DF, 1 nov. 2018, Seção 1, p. 90.

DIVYA, S.; SUDHA, M. Deep learning driven colorectal polyp analysis: A review of detection, classification and segmentation methods. **Frontiers in Artificial Intelligence**, v. 9, p. 1786679, 2026.

Docker Inc. **Docker Documentation**. 2026. Disponível em: <<https://docs.docker.com/>>.

DONG, B.; WANG, W.; FAN, D.-P.; LI, J.; FU, H.; SHAO, L. Polyp-PVT: Polyp segmentation with pyramid vision transformers. **CAAI Artificial Intelligence Research**, v. 2, p. 9150015, 2023.

DROŃ, M.; SZANDAŁA, T. Web application state management: A review of leading react frameworks. **IEEE Software**, v. 43, n. 2, p. 42–51, 2026.

EBAD, S. A.; ALHASHMI, A.; AMARA, M.; MILED, A. B.; SAQIB, M. Artificial intelligence-based software as a medical device (ai-samd): A systematic review. **Healthcare**, MDPI, v. 13, n. 7, p. 817, 2025. ISSN 2227-9032. Disponível em: <<https://doi.org/10.3390/healthcare13070817>>.

ELHADDAD, M.; HAMAM, S. Ai-driven clinical decision support systems: An ongoing pursuit of potential. **Cureus**, Cureus, v. 16, n. 4, p. e57728, 2024. ISSN 2168-8184. Disponível em: <<https://doi.org/10.7759/cureus.57728>>.

FATI, S. M.; SENAN, E. M.; AZAR, A. T. Hybrid and deep learning approach for early diagnosis of lower gastrointestinal diseases. **Sensors**, MDPI, v. 22, n. 11, p. 4079, 2022. ISSN 1424-8220.

FERLAY, J.; ERVIK, M.; LAM, F.; LAVERSANNE, M.; COLOMBET, M.; MERY, L.; PIÑEROS, M.; ZNAOR, A.; SOERJOMATARAM, I.; BRAY, F. **Global Cancer Observatory: Cancer Today**. Lyon, France: [s.n.], 2024. GLOBOCAN 2022 World Fact Sheet. Disponível em: <<https://gco.iarc.who.int/today>>.

GANGRADE, S.; SHARMA, P. C.; SHARMA, A. K.; GANGRADE, J. Computer-aided polyps classification from colonoscopy using stacking-based deep learning model. **Brazilian Archives of Biology and Technology**, Instituto de Tecnologia do Paraná (TECPAR), v. 68, p. e25240870, 2025. ISSN 1678-4324. Disponível em: <<https://doi.org/10.1590/1678-4324-2025240870>>.

GELU-SIMEON, M.; MAMOU, A.; SAINT-GEORGES, G.; ALEXIS, M.; SAUTEREAU, M.; MAMOU, Y.; SIMEON, J. Deep learning model applied to real-time delineation of colorectal polyps. **BMC Medical Informatics and Decision Making**, Springer Nature, v. 25, n. 206, p. 1–14, 2025. ISSN 1472-6947. Disponível em: <<https://doi.org/10.1186/s12911-025-03047-y>>.

GitHub. **About releases**. 2026. <<https://docs.github.com/en/repositories/releasing-projects-on-github/about-releases>>. Acesso em: 19 jul. 2026.

GONÇALVES, J. V. L.; SOUZA, D. V. d.; SANTOS, C. I. A. T. d.; NASCIMENTO, C. E. T. d.; CRUZ, L. B. d.; JUNIOR, D. A. D.; DINIZ, J. a. O. B. Diagnóstica: Ferramenta cadx para diagnóstico de doenças pulmonares em imagens radiológicas. In: **Anais do XXIV Simpósio Brasileiro de Computação Aplicada à Saúde (SBCAS)**. [S.l.: s.n.], 2024. p. 214–225.

GONG, E. J.; BANG, C. S.; LEE, J. J.; JEONG, H. M.; BAIK, G. H.; JEONG, J. H.; DICK, S.; LEE, G. H. Clinical decision support system for all stages of gastric carcinogenesis in real-time endoscopy: Model establishment and validation study. **Journal of Medical Internet Research**, v. 25, p. e50448, 2023.

HASHASH, J. G.; NG, F. Y. C.; FARRAYE, F. A.; WANG, Y.; COLUCCI, D. R.; BAXI, S.; MUNEER, S.; REDDAN, M.; SHINGRU, P.; MELMED, G. Y. Inter- and intraobserver variability on endoscopic scoring systems in crohn's disease and ulcerative colitis: A systematic review and meta-analysis. **Inflammatory Bowel Diseases**, Oxford University Press, v. 30, n. 11, p. 2217–2226, 2024. ISSN 1078-0998. Disponível em: <<https://doi.org/10.1093/ibd/izae051>>.

JEON, H. J.; KEUM, B.; JEONG, E. S.; KIM, S.-E.; MOON, C. M.; LEE, B.; KIM, S.; CHOI, H. S.; LEE, J. M.; KIM, E. S.; JEEN, Y. T. Clinical efficacy of real-time artificial intelligence-assisted colonoscopy in colorectal polyp detection: A prospective multicenter randomized controlled trial. **Gut and Liver**, v. 20, n. 1, p. 97–106, 2026. ISSN 1976-2283. Disponível em: <<https://doi.org/10.5009/gnl250369>>.

JIANG, L. Y.; LIU, X. C.; NEJATIAN, N. P.; NASIR-MOIN, M.; WANG, D.; ABIDIN, A.; EATON, K.; RIINA, H. A.; LAUFER, I.; PUNJABI, P.; MICELI, M.; KIM, N. C.; ORILLAC, C.; SCHNURMAN, Z.; LIVIA, C.; WEISS, H.; KURLAND, D.; NEIFERT, S.; DASTAGIRZADA, Y.; KONDZIOŁKA, D.; CHEUNG, A. T. M.; YANG, G.; CAO, M.; FLORES, M.; COSTA, A. B.; APHINYANAPHONGS, Y.; CHO, K.; OERMANN, E. K. Health system-scale language models are all-purpose prediction engines. **Nature**, Springer Nature, v. 619, n. 7970, p. 357–362, jul. 2023. ISSN 1476-4687. Disponível em: <<https://doi.org/10.1038/s41586-023-06160-y>>.

KHRYASHCHEV, V. V. Concept for a clinical decision support system in gastrointestinal endoscopy. **Measuring. Monitoring. Management. Control.**, n. 2, p. 137–144, 2025.

KUFEL, J.; BARGIEŁ-ŁĄCZEK, K.; KOCOT, S.; KOŹLIK, M.; BARTNIKOWSKA, W.; JANIK, M.; CZOGALIK, Ł.; DUDEK, P.; MAGIERA, M.; LIS, A.; PASZKIEWICZ, I.; NAWRAT, Z.; CEBULA, M.; GRUSZCZYŃSKA, K. What is machine learning, artificial neural networks and deep learning?—examples of practical applications in medicine. **Diagnostics**, MDPI, v. 13, n. 15, p. 2582, 2023. ISSN 2075-4418. Disponível em: <<https://doi.org/10.3390/diagnostics13152582>>.

LEHVÄ, J.; MÄKITALO, N.; MIKKONEN, T. Consumer-driven contract tests for microservices: A case study. In: **Product-Focused Software Process Improvement (PROFES 2019)**. [S.l.]: Springer, 2019. (Lecture Notes in Computer Science, v. 11915), p. 497–512.

MARELLA, V. Comparative analysis of container orchestration platforms: Kubernetes vs. Docker swarm. **International Journal of Scientific Research in Science and Technology**, v. 11, n. 5, p. 526–543, 2024.

MARY, A. A.; RAJ, P. W. A.; KARTHIK, C.; KARUNAHARAN, A. GastroNet: A custom deep learning approach for classification of anomalies in gastrointestinal endoscopy images. **Current Medical Imaging**, v. 20, 2024.

MATHEWS, S. C.; MCSHEA, M. J.; HANLEY, C. L.; RAVITZ, A.; LABRIQUE, A. B.; COHEN, A. B. Digital health: a path to validation. **npj Digital Medicine**, Nature Publishing Group, v. 2, n. 38, 2019.

MAURÍCIO, J.; DOMINGUES, I.; BERNARDINO, J. Comparing vision transformers and convolutional neural networks for image classification: A literature review.

Applied Sciences, MDPI, v. 13, n. 9, p. 5521, 2023. ISSN 2076-3417. Disponível em: <<https://doi.org/10.3390/app13095521>>.

MENNELLA, C.; MANISCALCO, U.; PIETRO, G. D.; ESPOSITO, M. Ethical and regulatory challenges of ai technologies in healthcare: A narrative review. **Heliyon**, Elsevier, v. 10, p. e26297, 2024. ISSN 2405-8440. Disponível em: <<https://doi.org/10.1016/j.heliyon.2024.e26297>>.

Meta Open Source. **React Documentation: Build user interfaces out of individual pieces**. 2026. Accessed: July 2026. Disponível em: <<https://react.dev>>.

MOHAMMED, F. A.; TUNE, K. K.; ASSEFA, B. G.; JETT, M.; MUHIE, S. Medical image classifications using convolutional neural networks: A survey of current methods and statistical modeling of the literature. **Machine Learning and Knowledge Extraction**, MDPI, v. 6, n. 1, p. 699–735, 2024.

MOORE, K. L.; DALLEY, A. F.; AGUR, A. M. R. **Anatomia orientada para a clínica**. 8. ed. Rio de Janeiro: Guanabara Koogan, 2019. ISBN 978-85-277-3459-2.

MUHAMMAD, D.; BENDECHACHE, M. Unveiling the black box: A systematic review of explainable artificial intelligence in medical image analysis. **Computational and Structural Biotechnology Journal**, Elsevier, v. 24, p. 542–560, 2024. ISSN 2001-0370. Disponível em: <<https://doi.org/10.1016/j.csbj.2024.08.021>>.

NANKYA, M.; MUGISA, A.; USMAN, Y.; UPADHYAY, A.; CHATAUT, R. Security and privacy in e-health systems: A review of ai and machine learning techniques. **IEEE Access**, IEEE, v. 12, p. 148796–148816, 2024. ISSN 2169-3536. Disponível em: <<https://doi.org/10.1109/ACCESS.2024.3469215>>.

NIJJAR, G. S.; AULAKH, S. K.; SINGH, R.; CHANDI, S. K. Emerging technologies in endoscopy for gastrointestinal neoplasms: A comprehensive overview. **Cureus**, v. 16, n. 6, p. e62946, 2024. Disponível em: <<https://doi.org/10.7759/cureus.62946>>.

NOGUEIRA-RODRÍGUEZ, A.; DOMÍNGUEZ-CARBAJALES, R.; LÓPEZ-FERNÁNDEZ, H.; IGLESIAS, ; CUBIELLA, J.; FDEZ-RIVEROLA, F.; REBOIRO-JATO, M.; GLEZ-PEÑA, D. Deep neural networks approaches for detecting and classifying colorectal polyps. **Neurocomputing**, v. 423, p. 721–734, 2021.

NOOR, M. N.; NAZIR, M.; ASHRAF, I.; ALMUJALLY, N. A.; ASLAM, M.; JILANI, S. F. Gastronet: A robust attention-based deep learning and cosine similarity feature selection framework for gastrointestinal disease classification from endoscopic images. **CAAI Transactions on Intelligence Technology**, John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology and Chongqing University of Technology, v. 8, n. 4, p. 1007–1020, 2023. ISSN 2468-2322. Disponível em: <<https://doi.org/10.1049/cit2.12231>>.

NOPOUR, R. Use of clinical decision support systems for diagnosis and prognosis of gastric cancer: A scoping review. **Health Science Reports**, v. 8, p. e71203, 2025.

ODOFIN, O. T.; ADEKUNLE, B. I.; OGBUEFI, E.; OGEAWUCHI, J. C.; ADANIGBO, O. S.; GBENLE, T. P. Improving healthcare data intelligence through custom NLP pipelines and FastAPI microservices. **Journal of Frontiers in Multidisciplinary Research**, v. 4, n. 1, p. 390–397, 2023.

OPENJA, M. Studying the practices of deploying machine learning projects on Docker. In: **Proceedings of the International Conference on Evaluation and Assessment in Software Engineering (EASE)**. [S.l.: s.n.], 2022.

PATHAN, S.; SHAHINI, A.; SALVI, M.; ALI, T.; LEWIS, J. R.; MOLINARI, F.; ACHARYA, U. R. Artificial intelligence in gastrointestinal disease diagnosis: A systematic review of endoscopy, histology, and radiology applications. **WIREs Data Mining and Knowledge Discovery**, Wiley, v. 16, p. e70074, 2026.

PINTO, H. A.; SANTANA, J. S. S.; CHIORO, A. Por uma transformação digital que assegure o direito à saúde e à proteção de dados pessoais. **Saúde em Redes**, v. 8, n. 2, p. 361–371, 2022. ISSN 2446-4813.

POGORELOV, K.; RANDEL, K. R.; GRIWODZ, C.; ESKELAND, S. L.; LANGE, T. de; JOHANSEN, D.; SPAMPINATO, C.; DANG-NGUYEN, D.-T.; LUX, M.; SCHMIDT, P. T.; RIEGLER, M.; HALVORSEN, P. Kvasir: A multi-class image dataset for computer-aided gastrointestinal disease detection. In: **Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys '17)**. Taipei, Taiwan: Association for Computing Machinery, 2017. p. 164–169. ISBN 9781450350020. Disponível em: <<https://doi.org/10.1145/3083187.3083212>>.

PRESSMAN, R. S.; MAXIM, B. R. **Engenharia de Software: Uma Abordagem Profissional**. 8. ed. Porto Alegre: AMGH Editora, 2016. Tradução de João Eduardo Nóbrega Tortello. Obra originalmente publicada como *Software Engineering: A Practitioner's Approach*, 8th Edition. ISBN 978-85-8055-534-9.

RAMÍREZ, S. **FastAPI: Modern, fast (high-performance), web framework for building APIs with Python**. 2018. Acesso em: 5 jul. 2026. Disponível em: <<https://fastapi.tiangolo.com>>.

REDDYKAVULURI, H. V. PostgreSQL vs. Oracle: A comparative study of performance, scalability, and enterprise adoption. **International Journal of Emerging Trends in Computer Science and Information Technology**, v. 5, n. 2, p. 33–41, 2024.

ROSS, R. G. **Business Rule Concepts: Getting to the Point of Knowledge**. 3. ed. Houston: Business Rule Solutions, LLC, 2005.

SALIMPARSA, M.; SEDIG, K.; LIZOTTE, D. J.; ABDULLAH, S. S.; CHALABIANLOO, N.; MUANDA, F. T. Explainable ai for clinical decision support systems: Literature review, key gaps, and research synthesis. **Informatics**, MDPI, v. 12, n. 4, p. 119, 2025. ISSN 2227-9709. Disponível em: <<https://doi.org/10.3390/informatics12040119>>.

SAMPAIO, F. A. V. C. Proposta de sistema web de apoio à decisão clínica baseado em aprendizado de máquina para análise de sobrevivência de pacientes com doenças cardiovasculares. In: **Anais do V Encontro de Computação do Oeste Potiguar (ECOP)**. Pau dos Ferros, RN, Brasil: Universidade Federal Rural do Semi-Árido (UFERSA), 2022. v. 6, p. 5–8. ISSN 2526-7574.

SELVARAJU, R. R.; COGSWELL, M.; DAS, A.; VEDANTAM, R.; PARIKH, D.; BATRA, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. **International Journal of Computer Vision**, Springer, v. 128, n. 2, p. 336–359, 2020.

SHAUKAT, A.; LICHTENSTEIN, D. R.; SOMERS, S. C.; CHUNG, D. C.; PERDUE, D. G.; GOPAL, M.; COLUCCI, D. R.; PHILLIPS, S. A.; MARKA, N. A.; CHURCH, T. R.; BRUGGE, W. R. Computer-aided detection improves adenomas per colonoscopy for screening and surveillance colonoscopy: A randomized trial. **Gastroenterology**, v. 163, n. 3, p. 732–741, 2022. ISSN 0016-5085.

SHINOZAKI, S.; WATANABE, J.; KANNO, T.; NAKAZAWA, K.; YANO, T. Comparative performance of artificial intelligence-based computer-aided detection systems for colorectal polyps: A systematic review and network meta-analysis. **Digestive Endoscopy**, John Wiley & Sons Australia, Ltd on behalf of Japan Gastroenterological Endoscopy Society, v. 38, p. e70138, 2026. Disponível em: <<https://doi.org/10.1111/den.70138>>.

SILVA, D.; RAFAEL, J.; FONTE, A. Toward optimal virtualization: An updated comparative analysis of Docker and LXD container technologies. **Computers**, v. 13, n. 4, p. 94, 2024.

SIVARI, E.; BOSTANCI, E.; GUZEL, M. S.; ACICI, K.; ASUROGLU, T.; AYYILDIZ, T. E. A new approach for gastrointestinal tract findings detection and classification: Deep learning-based hybrid stacking ensemble models. **Diagnostics**, MDPI, v. 13, n. 4, p. 720, 2023. ISSN 2075-4418. Disponível em: <<https://doi.org/10.3390/diagnostics13040720>>.

SOMMERVILLE, I. **Engenharia de Software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

SOUZA, T. R. d.; LOPES, B. L.; CHAHLA, S. M.; AGUIAR, C. L.; CABRAL, J. R. L. G. Endoscopia digestiva alta: principais indicações. **Revista Ibero-Americana de Humanidades, Ciências e Educação**, v. 10, n. 8, p. 4056–4063, 2024. ISSN 2675-3375.

SRINIVASU, P. N.; SANDHYA, N.; JHAVERI, R. H.; RAUT, R. From blackbox to explainable ai in healthcare: Existing tools and case studies. **Mobile Information Systems**, Hindawi, v. 2022, p. 1–20, 2022.

STOUMPOS, A. I.; KITSIOS, F.; TALIAS, M. A. Digital transformation in healthcare: Technology acceptance and its applications. **International Journal of Environmental Research and Public Health**, MDPI, v. 20, n. 4, p. 3407, 2023. ISSN 1660-4601. Disponível em: <<https://doi.org/10.3390/ijerph20043407>>.

SULLIVAN, B. A.; NOUJAIM, M.; ROPER, J. Cause, epidemiology, and histology of polyps and pathways to colorectal cancer. **Gastrointestinal Endoscopy Clinics of North America**, v. 32, n. 2, p. 177–194, 2022.

TALEBI, M.; BOZORGCHAMI, N.; MISHRA, G.; MISHRA, G.; GHALE, R. A. Clinical validation of artificial intelligence for gastrointestinal diseases. **Informatics in Medicine Unlocked**, Elsevier, v. 61, p. 101736, 2026. ISSN 2352-9148. Disponível em: <<https://doi.org/10.1016/j.imu.2026.101736>>.

TAN, M.; LE, Q. V. Efficientnet: Rethinking model scaling for convolutional neural networks. In: **Proceedings of the 36th International Conference on Machine Learning (ICML)**. Long Beach, California, USA: PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 6105–6114. Disponível em: <<https://proceedings.mlr.press/v97/tan19a.html>>.

The PostgreSQL Global Development Group. **PostgreSQL 17.5 Documentation**. 2025. Disponível em: <<https://www.postgresql.org/docs/17/>>.

TORAB-MIANDOAB, A.; SAMAD-SOLTANI, T.; JODATI, A.; REZAEI-HACHESU, P. Interoperability of heterogeneous health information systems: A systematic literature review. **BMC Medical Informatics and Decision Making**, BioMed Central, v. 23, n. 1, p. 18, 2023. ISSN 1472-6947. Disponível em: <<https://doi.org/10.1186/s12911-023-02115-5>>.

URNIKIENĖ, J.; STEPONAVIČIENĖ, V.; ATANASOV, S. Comparative read performance analysis of PostgreSQL and MongoDB in e-commerce: An empirical study of filtering and analytical queries. **Big Data and Cognitive Computing**, v. 10, n. 2, p. 66, 2026.

VAISHYA, R.; GOENKA, M. K.; SIBAL, A.; KAR, S.; REDDY, S. Artificial intelligence in gastroenterology: A comprehensive review. **National Board of Examinations Journal of Medical Sciences**, v. 3, n. 6, p. 729–745, jun. 2025. Disponível em: <<https://doi.org/10.61770/NBEJMS.2025.v03.i06.011>>.

VELDEN, B. H. M. van der; KUIJF, H. J.; GILHUIJS, K. G. A.; VIERGEVER, M. A. Explainable artificial intelligence (xai) in deep learning-based medical image analysis. **Medical Image Analysis**, Elsevier, v. 79, p. 102470, 2022. ISSN 1361-8415. Disponível em: <<https://doi.org/10.1016/j.media.2022.102470>>.

VIANA, P. da S. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação), **Diagnósticos de Anomalias Gastrointestinais em Imagens Endoscópicas Usando Ensemble Stacking de CNNs e Vision Transformers**. Juazeiro do Norte, CE, Brasil: [s.n.], 2026.

VIANA, P. da S.; CRUZ, L. B. da; DINIZ, J. O. B.; SANDES, N. C. Ensemble stacking de cnns e vision transformers para classificação de anormalidades em imagens endoscópicas. In: **Anais do Congresso da Sociedade Brasileira de Computação**. [S.l.: s.n.], 2026.

WANG, W.; XIE, E.; LI, X.; FAN, D.-P.; SONG, K.; LIANG, D.; LU, T.; LUO, P.; SHAO, L. Pvt v2: Improved baselines with pyramid vision transformer. **Computational Visual Media**, Springer, v. 8, n. 3, p. 415–424, 2022. ISSN 2096-0662. Disponível em: <<https://doi.org/10.1007/s41095-022-0270-1>>.

WENG, Y.; WU, J. Database management systems for artificial intelligence: Comparative analysis of PostgreSQL and MongoDB. **World Journal of Advanced Research and Reviews**, v. 25, n. 2, p. 2336–2342, 2025.

WU, Z.; LV, F.; CHEN, C.; HAO, A.; LI, S. Colorectal polyp segmentation in the deep learning era: A comprehensive survey. **arXiv preprint arXiv:2401.11734**, 2024. Disponível em: <<https://arxiv.org/abs/2401.11734>>.

ZHEN, C.; YAO, C.; LI, S.; LIN, Z.; IBRAHIM, U. M.; CHIPUSU, K.; ZHENG, B.; LIANG, R. Grad-cam based deep learning analytics for image-level colon disease classification based on graph neural networks and vision transformers. **Frontiers in Physiology**, Frontiers Media SA, v. 17, p. 1734299, 2026. ISSN 1664-042X. Disponível em: <<https://doi.org/10.3389/fphys.2026.1734299>>.

APÊNDICE A – REQUISITOS FUNCIONAIS

Os requisitos funcionais descrevem as funcionalidades que o sistema ClinicAI deve oferecer aos seus usuários, considerando os perfis de acesso definidos para a aplicação (Subseções 4.3.1, 4.3.2 e 4.3.3). Os requisitos foram organizados por módulos funcionais, abrangendo autenticação, gerenciamento de usuários, clínicas, pacientes, exames, análise automatizada por IA, revisão médica, auditoria e *dashboard*.

A.1 Autenticação e Controle de Acesso

O sistema deve permitir a autenticação segura de usuários por meio de login (e-mail único) e senha, utilizando *tokens* JWT para controle de sessão e autorização de acesso às rotas da aplicação e um mecanismo de renovação de sessão utilizando *refresh tokens*, além do RBAC, permitindo diferentes níveis de acesso ao sistema. A Tabela 5 descreve os requisitos funcionais referentes à autenticação e ao controle de acesso.

Tabela 5 – Requisitos funcionais de autenticação e controle de acesso.

ID	Nome	Descrição
RF01	Autenticar usuário	Permitir que usuários cadastrados acessem o sistema por meio de e-mail e senha válidos.
RF02	Encerrar sessão	Permitir que o usuário autenticado encerre sua sessão no sistema de forma segura.
RF03	Manter sessão autenticada	Permitir que o usuário permaneça autenticado durante o período de validade da sessão, por meio de renovação via <i>refresh token</i> .
RF04	Controlar acesso por perfil	Restringir o acesso às funcionalidades do sistema conforme o perfil do usuário autenticado.
RF05	Bloquear acesso não autorizado	Impedir que usuários não autenticados ou sem permissão acessem rotas e funcionalidades protegidas.

Fonte: Elaborado pela autora.

A.2 Gerenciamento de Usuários

O sistema deve permitir o gerenciamento de usuários, incluindo cadastro, edição, ativação, inativação e associação com clínicas e perfis de acesso. O perfil ADM tem permissão para realizar todas essas funcionalidades, enquanto o GDC pode cadastrar, consultar, editar, ativar, inativar e redefinir a senha de usuários com perfil MED vinculados à sua própria clínica, e o MED pode apenas editar os dados do seu próprio perfil. A Tabela 6 descreve os requisitos funcionais referentes ao gerenciamento de usuários.

Tabela 6 – Requisitos funcionais de gerenciamento de usuários.

ID	Nome	Descrição
RF06	Cadastrar usuário	Permitir o cadastro de usuários com informações pessoais, credenciais de acesso, perfil e vínculo institucional.
RF07	Consultar usuários	Permitir a listagem, busca, filtragem e visualização detalhada dos usuários cadastrados no sistema.
RF08	Atualizar usuário	Permitir a edição das informações cadastrais de usuários previamente registrados.
RF09	Editar o próprio perfil	Permitir que usuários autenticados alterem informações do próprio perfil, respeitando suas permissões.
RF10	Ativar ou inativar usuário	Permitir o controle do status do usuário para utilização ou impedimento de acesso às operações do sistema sem remoção permanente de seus dados.
RF11	Redefinir senha de usuário	Permitir a redefinição de senha de usuários cadastrados no sistema.
RF12	Associar usuário à clínica	Permitir o vínculo de um usuário a uma clínica cadastrada, conforme sua função no sistema.
RF13	Associar usuário a perfil de acesso	Permitir a atribuição de perfis de acesso aos usuários de acordo com suas responsabilidades no sistema.

Fonte: Elaborado pela autora.

A.3 Gerenciamento de Clínicas

O sistema deve permitir o gerenciamento de clínicas, incluindo cadastro, edição, ativação, inativação e associação com usuários MED e/ou GDC. O ADM tem permissão para realizar todas essas funcionalidades sobre qualquer clínica cadastrada, enquanto o MED e GDC podem consultar e atualizar os dados institucionais apenas da própria clínica à qual estão vinculados. A clínica tem preenchimento automático de endereço pelo Código de Endereçamento Postal (CEP) e validação de Cadastro Nacional de Pessoas Jurídicas (CNPJ). A Tabela 7 descreve os requisitos funcionais referentes ao gerenciamento de clínicas.

Tabela 7 – Requisitos funcionais de gerenciamento de clínicas.

ID	Nome	Descrição
RF14	Cadastrar clínica	Permitir o cadastro de clínicas com informações de identificação, dados de contato e endereço.
RF15	Consultar clínicas	Permitir a listagem, busca, filtragem e visualização detalhada das clínicas cadastradas no sistema.
RF16	Atualizar clínica	Permitir a edição das informações cadastrais das clínicas previamente registradas.
RF17	Ativar ou inativar clínica	Permitir o controle do status da clínica para utilização ou impedimento de acesso às operações do sistema sem remoção permanente de seus dados.
RF18	Validar dados da clínica	Verificar informações cadastrais da clínica, como CNPJ e dados obrigatórios, evitando inconsistências e duplicidades.
RF19	Preencher endereço por CEP	Permitir o preenchimento automático do endereço da clínica a partir do CEP informado.

Fonte: Elaborado pela autora.

A.4 Gerenciamento de Pacientes

O módulo deve permitir o gerenciamento de pacientes, incluindo cadastro, edição, associação dos pacientes com os médicos das clínicas às quais estão vinculados. O sistema deve realizar validações na integridade dos dados dos pacientes, com validação de Cadastro de Pessoas Físicas (CPF) e busca automática de endereço pelo CEP. O ADM tem permissão para realizar todas essas funcionalidades sobre qualquer paciente cadastrado, o GDC pode realizá-las para todos os pacientes vinculados à sua clínica, enquanto e o MED pode realizá-las apenas para os pacientes sob sua própria responsabilidade clínica. A Tabela 8 descreve os requisitos funcionais referentes ao gerenciamento de pacientes.

Tabela 8 – Requisitos funcionais de gerenciamento de pacientes.

ID	Nome	Descrição
RF20	Cadastrar paciente	Permitir o cadastro de pacientes com informações pessoais, clínicas, endereço e dados de contato.
RF21	Consultar pacientes	Permitir a listagem, busca, filtragem e visualização detalhada dos pacientes cadastrados.
RF22	Atualizar paciente	Permitir a edição das informações cadastrais e clínicas de pacientes previamente registrados.
RF23	Ativar ou inativar paciente	Permitir o controle do status do paciente sem remoção permanente de seus dados.
RF24	Validar dados do paciente	Verificar informações cadastrais do paciente, como CPF e campos obrigatórios, evitando inconsistências e duplicidades.
RF25	Preencher endereço por CEP	Permitir o preenchimento automático do endereço do paciente a partir do CEP informado.
RF26	Associar paciente à clínica	Permitir o vínculo do paciente a uma clínica cadastrada no sistema.
RF27	Associar paciente ao médico responsável	Permitir o vínculo do paciente ao médico responsável por seu acompanhamento clínico.
RF28	Consultar histórico do paciente	Permitir a visualização do histórico de exames e análises associados ao paciente.

Fonte: Elaborado pela autora.

A.5 Gerenciamento de Exames

O sistema deve permitir o gerenciamento de exames médicos dos pacientes cadastrados, incluindo *upload* de arquivos, armazenando informações clínicas, controlando o fluxo de processamento do exame e integrando-se com o módulo de IA. O fluxo dos exames contempla diferentes estados clínicos, permitindo o acompanhamento do processamento da análise automatizada e foi melhor detalhado na Subseção 5.5. A Tabela 9 descreve os requisitos funcionais referentes ao gerenciamento de exames.

Tabela 9 – Requisitos funcionais de gerenciamento de exames.

ID	Nome	Descrição
RF29	Cadastrar exame	Permitir o cadastro de exames vinculados a pacientes, clínicas e médicos responsáveis.
RF30	Enviar imagem do exame	Permitir o <i>upload</i> de imagens médicas associadas ao exame cadastrado.
RF31	Visualizar lista de exames	Permitir a listagem, busca e filtragem dos exames cadastrados.
RF32	Consultar exames	Permitir a visualização detalhada dos exames cadastrados, incluindo dados clínicos e administrativos.
RF33	Atualizar dados do exame	Permitir a edição de informações clínicas e administrativas do exame enquanto estiver no status “Pendente”.
RF34	Acompanhar status do exame	Permitir o acompanhamento do estado do exame durante seu fluxo de processamento.
RF35	Baixar arquivo do exame	Permitir o <i>download</i> da imagem ou dos arquivos associados ao exame.
RF36	Cancelar exame	Permitir o cancelamento de exames, mantendo seus dados armazenados para fins de histórico e rastreabilidade.
RF37	Consultar histórico do exame	Permitir a visualização das alterações de status e eventos relacionados ao exame ao longo do seu ciclo de vida.
RF38	Revisar resultado da análise do exame	Permitir a avaliação crítica do resultado gerado pela IA antes da conclusão do exame.
RF39	Registrar parecer médico	Permitir o registro de observações, apontamentos ou conclusão clínica relacionados ao exame analisado.
RF40	Concluir exame	Permitir a finalização do exame após a revisão do resultado e o registro do parecer clínico.
RF41	Imprimir relatório do exame	Permitir a geração de um relatório em PDF do exame já concluído, contendo os dados do paciente, do exame e o resultado diagnóstico registrado, para entrega ao paciente.

Fonte: Elaborado pela autora.

A.6 Análise de Exames com IA

O sistema deve integrar um serviço de IA responsável pelo processamento de imagens médicas endoscópicas e colonoscópicas. O módulo deve permitir pré-processamento das imagens, execução de inferência automática, geração de classificações, cálculo de confiança da predição e explicabilidade utilizando Grad-CAM. A Tabela 10 descreve os requisitos funcionais referentes à análise de exames com IA.

Tabela 10 – Requisitos funcionais de análise de exames com IA.

ID	Nome	Descrição
RF42	Submeter exame à análise por IA	Permitir que exames com imagem válida sejam enviados ao módulo de IA para processamento automatizado.
RF43	Processar imagem automaticamente	Executar o fluxo automatizado de preparação, inferência e análise da imagem médica enviada.
RF44	Classificar imagem do exame	Permitir a classificação automatizada da imagem como normal ou anormal, conforme as classes definidas pelo modelo utilizado.
RF45	Gerar resultado da análise	Gerar o resultado da inferência contendo classificação predita, nível de confiança e informações associadas ao processamento.
RF46	Gerar visualização Grad-CAM	Gerar mapa de ativação Grad-CAM para destacar visualmente as regiões da imagem consideradas relevantes pelo modelo.
RF47	Armazenar resultado da análise	Registrar os resultados produzidos pela IA e vinculá-los ao exame correspondente.
RF48	Consultar resultado da IA	Permitir a visualização do resultado da análise automatizada, incluindo classificação, confiança e Grad-CAM.
RF49	Restaurar exame cancelado ou com falha	Permitir a restauração de exames cancelados ou com falha no processamento, retornando-os ao status inicial para nova submissão à análise por IA.
RF50	Consultar histórico de análises de IA	Permitir a visualização das análises automatizadas realizadas, incluindo data, status, modelo utilizado e resultado obtido.

Fonte: Elaborado pela autora.

A.7 Auditoria e Rastreabilidade

O sistema deve registrar operações críticas realizadas na plataforma visando rastreabilidade, auditoria e controle das ações administrativas, clínicas e computacionais. O ADM tem permissão para realizar todas essas funcionalidades. A Tabela 11 descreve os requisitos funcionais referentes à auditoria e à rastreabilidade do sistema.

Tabela 11 – Requisitos funcionais de auditoria e rastreabilidade.

ID	Nome	Descrição
RF51	Registrar operações críticas	Registrar ações relevantes realizadas no sistema, incluindo autenticação, alterações cadastrais, envio de arquivos, processamento de exames e revisão médica.
RF52	Identificar usuário responsável e IP da máquina	Associar cada operação registrada ao usuário responsável pela ação e o endereço IP da máquina utilizada.
RF53	Registrar data e hora das operações	Armazenar data e horário das operações relevantes realizadas no sistema.
RF54	Consultar <i>logs</i> de auditoria	Permitir a visualização dos <i>logs</i> de auditoria armazenados.
RF55	Filtrar <i>logs</i> de auditoria	Permitir a filtragem dos <i>logs</i> de auditoria por usuário, operação, entidade, período ou tipo de evento.

Fonte: Elaborado pela autora.

A.8 *Dashboard* e Monitoramento

O sistema deve disponibilizar *dashboards* administrativos contendo indicadores gerais relacionados à utilização da plataforma. A Tabela 12 descreve os requisitos funcionais referentes à *dashboard* e ao monitoramento do sistema.

Tabela 12 – Requisitos funcionais de *dashboard* e monitoramento.

ID	Nome	Descrição
RF56	Visualizar indicadores gerais da plataforma	Apresentar indicadores quantitativos de usuários, clínicas, pacientes e exames cadastrados.
RF57	Monitorar exames por status	Exibir a distribuição dos exames conforme seus estados no fluxo operacional do sistema.
RF58	Visualizar indicadores das análises de IA	Apresentar informações e métricas das análises automatizadas realizadas, incluindo resultados, falhas e volume de processamento.
RF59	Restringir indicadores por escopo de acesso	Apresentar os indicadores do <i>dashboard</i> automaticamente restritos ao escopo de atuação do perfil autenticado: o MED visualiza indicadores relativos aos seus próprios pacientes e exames, o GDC visualiza indicadores operacionais da própria clínica, e o ADM visualiza indicadores agregados de toda a plataforma.

Fonte: Elaborado pela autora.

APÊNDICE B – REQUISITOS NÃO-FUNCIONAIS

Os requisitos não-funcionais descrevem as propriedades, restrições técnicas e atributos de qualidade que orientam o funcionamento do sistema ClinicAI. Esses requisitos abrangem aspectos de segurança, privacidade, persistência, modularização, desempenho, escalabilidade, portabilidade, manutenibilidade e usabilidade, considerando que o sistema manipula dados clínicos sensíveis e realiza processamento automatizado de imagens médicas por meio de IA.

B.1 Segurança

A segurança do sistema visa garantir a confidencialidade, a integridade e a disponibilidade das informações. Para isso, o sistema utiliza diversos mecanismos para proteção, a fim de minimizar as vulnerabilidades e assegurar a proteção dos dados clínicos processados pelo sistema. A Tabela 13 descreve os requisitos não-funcionais referentes à segurança da aplicação.

Tabela 13 – Requisitos não-funcionais de segurança.

Código	Descrição
RNF01	O sistema deve utilizar autenticação segura para controlar o acesso dos usuários à aplicação, utilizando <i>tokens</i> para controle de sessão e autorização de requisições.
RNF02	O sistema deve proteger rotas privadas, permitindo acesso apenas a usuários autenticados e autorizados.
RNF03	O sistema deve controlar permissões por meio de perfis de acesso baseados em RBAC, aplicando o princípio do menor privilégio.
RNF04	O sistema deve armazenar senhas de forma segura, utilizando mecanismos de criptografia ou <i>hash</i> adequados.
RNF05	O sistema deve validar os dados enviados à API antes da execução das operações.
RNF06	O sistema deve restringir o acesso a imagens, exames, resultados e dados clínicos apenas a usuários autorizados.
RNF07	O sistema deve registrar operações críticas para fins de auditoria, rastreabilidade e responsabilização.
RNF08	O sistema deve considerar princípios de proteção de dados sensíveis, em conformidade com boas práticas de segurança e privacidade aplicáveis a sistemas de saúde.

Fonte: Elaborado pela autora.

B.2 Persistência de Dados

A persistência de dados é responsável por garantir o armazenamento permanente das informações geradas durante a utilização do sistema. Essa abordagem assegura a integridade dos dados, reduz o risco de perdas decorrentes de falhas do sistema e possibilita consultas, atualizações e gerenciamento eficiente das informações armazenadas. A Tabela 14 descreve os requisitos não-funcionais referentes à persistência de dados.

Tabela 14 – Requisitos não-funcionais de persistência de dados.

Código	Descrição
RNF09	O sistema deve utilizar um banco de dados relacional, sendo o PostgreSQL o sistema gerenciador do banco.
RNF10	O sistema deve garantir integridade relacional entre usuários, clínicas, pacientes, exames e análises de IA.
RNF11	O sistema deve impedir inconsistências em vínculos obrigatórios entre entidades, como paciente, clínica, médico responsável e exame.
RNF12	O sistema deve versionar a estrutura do banco de dados por meio da ferramenta de migração Alembic.
RNF13	O sistema deve manter histórico de processamento dos exames e das análises automatizadas realizadas.
RNF14	O sistema deve preservar os dados clínicos e administrativos por meio de inativação lógica quando aplicável, evitando exclusões físicas desnecessárias.

Fonte: Elaborado pela autora.

B.3 Modularização e Organização Arquitetural

O sistema foi desenvolvido seguindo uma arquitetura modular, na qual suas funcionalidades são organizadas em componentes independentes e de responsabilidades bem definidas, separando o *frontend*, o *backend*, o banco de dados e o módulo de IA em serviços distintos, facilitando o desenvolvimento, a manutenção e a evolução da aplicação, além de possibilitar que novas funcionalidades sejam incorporadas sem impactar significativamente as demais. A Tabela 15 descreve os requisitos não-funcionais referentes à modularização.

Tabela 15 – Requisitos não-funcionais de modularização e organização arquitetural.

Código	Descrição
RNF15	O sistema deve organizar o <i>backend</i> em módulos independentes e de responsabilidade definida.
RNF16	O sistema deve separar as camadas de modelos (<i>models</i>), esquemas (<i>schemas</i>), rotas (<i>routes</i>) e serviços (<i>services</i>) da aplicação.
RNF17	O sistema deve concentrar as regras de negócio na camada de serviços do <i>backend</i> .
RNF18	O sistema deve separar os serviços de <i>frontend</i> , <i>backend</i> , banco de dados e IA.
RNF19	O sistema deve permitir que o módulo de IA funcione como serviço independente da API principal.
RNF20	O sistema deve padronizar a comunicação entre seus componentes por meio de REST API.
RNF21	O sistema deve favorecer a reutilização de componentes, módulos e serviços em futuras expansões da plataforma.

Fonte: Elaborado pela autora.

B.4 Desempenho e Confiabilidade

O sistema deve apresentar comportamento estável durante as operações de cadastro, consulta, *upload* e processamento automatizado de exames. Considerando o custo computacional associado à análise de imagens médicas, o ClinicAI deve evitar reprocessamentos desnecessários, registrar falhas e preservar os resultados já obtidos. A Tabela 16 descreve os requisitos não-funcionais referentes ao desempenho e à confiabilidade.

Tabela 16 – Requisitos não-funcionais de desempenho e confiabilidade.

Código	Descrição
RNF22	O sistema deve responder de forma adequada às operações de cadastro, consulta e atualização realizadas pelos usuários.
RNF23	O sistema deve manter a estabilidade da aplicação durante o envio e processamento de imagens médicas.
RNF24	O sistema deve armazenar os resultados das análises automatizadas para permitir consultas posteriores sem necessidade de novo processamento.
RNF25	O sistema deve registrar falhas ocorridas durante o processamento das imagens pela IA.
RNF26	O sistema deve permitir a identificação do status de processamento dos exames.
RNF27	O sistema deve evitar perda de dados em caso de falha durante o processamento automatizado.
RNF28	O sistema deve permitir a recuperação do fluxo operacional de exames em caso de erro ou falha de processamento.

Fonte: Elaborado pela autora.

B.5 Escalabilidade

A escalabilidade refere-se à capacidade do sistema de suportar o aumento de usuários, exames processados e volume de dados sem comprometer seu desempenho. A utilização de uma arquitetura baseada em serviços independentes e containerizados permite que cada componente da aplicação possa ser expandido de forma individual conforme a demanda. Dessa forma, o sistema pode acomodar futuras ampliações, como o processamento simultâneo de múltiplos exames, a inclusão de novos modelos de IA ou o aumento da infraestrutura computacional disponível. A Tabela 17 descreve os requisitos não-funcionais referentes à escalabilidade.

Tabela 17 – Requisitos não-funcionais de escalabilidade.

Código	Descrição
RNF29	O sistema deve permitir a expansão gradual dos módulos da plataforma.
RNF30	O sistema deve permitir a atualização, substituição ou inclusão de novos modelos de IA.
RNF31	O sistema deve desacoplar o processamento de IA da API principal.
RNF32	O sistema deve permitir processamento assíncrono de análises automatizadas em versões futuras do sistema.
RNF33	O sistema deve permitir escalabilidade horizontal dos serviços da arquitetura.

Fonte: Elaborado pela autora.

B.6 Portabilidade

A portabilidade do sistema foi alcançada por meio da utilização da plataforma Docker, que possibilita empacotar todos os componentes da aplicação juntamente com suas dependências em *containers*. Essa abordagem reduz problemas relacionados às diferenças entre ambientes de desenvolvimento, teste e produção, garantindo maior consistência durante a implantação do sistema, podendo ser executada em diferentes sistemas operacionais ou infraestruturas computacionais, desde que possuam suporte à execução de *containers* Docker. A Tabela 18 descreve os requisitos não-funcionais referentes à portabilidade.

Tabela 18 – Requisitos não-funcionais de portabilidade.

Código	Descrição
RNF34	O sistema deve executar a aplicação utilizando contêineres Docker.
RNF35	O sistema deve orquestrar os serviços da plataforma utilizando Docker Compose.
RNF36	O sistema deve permitir execução independente dos serviços da aplicação.

Fonte: Elaborado pela autora.

B.7 Manutenibilidade

A manutenibilidade consiste na facilidade de modificar, corrigir ou evoluir o sistema ao longo de seu ciclo de vida, por meio da adoção de boas práticas de desenvolvimento, como a organização do código em módulos, separação entre as camadas da aplicação, padronização da estrutura do projeto e utilização de tecnologias amplamente documentadas. Essa organização facilita a identificação de falhas, a implementação de novas funcionalidades e a atualização dos componentes existentes, reduzindo o esforço necessário para futuras manutenções. A Tabela 19 descreve os requisitos não-funcionais referentes à manutenibilidade.

Tabela 19 – Requisitos não-funcionais de manutenibilidade.

Código	Descrição
RNF37	O sistema deve facilitar a manutenção e a continuidade do projeto por meio de arquitetura modular.
RNF38	O sistema deve permitir a atualização independente dos serviços do sistema.
RNF39	O sistema deve facilitar a evolução futura da plataforma e a integração de novos módulos.

Fonte: Elaborado pela autora.

B.8 Usabilidade

A usabilidade busca proporcionar uma interação simples, intuitiva e eficiente entre o usuário e o sistema. A interface, portanto, foi desenvolvida utilizando o *framework* CoreUI para React, oferecendo uma organização visual padronizada, menus intuitivos e componentes gráficos que facilitam a navegação. O fluxo de utilização foi planejado para reduzir a quantidade de etapas necessárias para a realização das principais operações, como o envio de exames, a visualização dos resultados e o acesso ao histórico de análises, proporcionando uma experiência de uso mais eficiente e acessível aos profissionais que utilizarão a plataforma. A Tabela 20 descreve os requisitos não-funcionais de usabilidade.

Tabela 20 – Requisitos não-funcionais de usabilidade.

Código	Descrição
RNF40	O sistema deve utilizar interface responsiva compatível com diferentes resoluções de tela.
RNF41	O sistema deve facilitar a navegação entre módulos administrativos e clínicos do sistema.
RNF42	O sistema deve disponibilizar <i>dashboards</i> com informações organizadas e de fácil interpretação.

Fonte: Elaborado pela autora.

APÊNDICE C – REGRAS E NEGÓCIO

Essas regras de negócio foram definidas para garantir a integridade das informações e a correta execução do fluxo de análise de exames, levando em consideração os requisitos funcionais e não-funcionais, estabelecendo limites para as ações do sistema, de forma que estejam de acordo com regras organizacionais e administrativas de ambientes médicos conhecidos. As regras foram organizadas por categoria temática, abrangendo cadastro e vínculos, fluxo de processamento de exames, análise de IA e segurança, auditoria e controle de acesso.

C.1 Cadastro e Vínculos de Pacientes e Exames

Estas regras garantem a integridade dos vínculos estabelecidos entre pacientes, clínicas, médicos e exames no momento do cadastro, assegurando a consistência dos dados cadastrais associados a cada entidade do sistema. A Tabela 21 descreve as regras de negócio referentes ao cadastro e aos vínculos de pacientes e exames.

Tabela 21 – Regras de negócio de cadastro e vínculos de pacientes e exames.

Código	Descrição
RN01	O exame deve possuir obrigatoriamente um arquivo de imagem válido no momento do cadastro, garantindo que toda análise de IA sempre parta de uma imagem existente.
RN02	Apenas arquivos nos formatos JPG, JPEG e PNG podem ser enviados ao sistema.
RN03	Cada paciente deve estar vinculado a uma clínica cadastrada no sistema.
RN04	Cada paciente deve possuir um médico responsável vinculado ao seu acompanhamento clínico.
RN05	O exame deve herdar automaticamente a clínica vinculada ao paciente associado.
RN06	O exame deve herdar automaticamente o médico responsável vinculado ao paciente associado.
RN07	O CPF do paciente deve ser único dentro da mesma clínica.

Fonte: Elaborado pela autora.

C.2 Fluxo de Processamento de Exames

Estas regras definem o ciclo de vida do exame no ClinicAI, estabelecendo os status possíveis, as transições permitidas entre eles e as responsabilidades de cada perfil de usuário ao longo do fluxo de análise e revisão (Subseção 4.3.5). A Tabela 22 descreve as regras de negócio referentes ao fluxo de processamento de exames.

Tabela 22 – Regras de negócio do fluxo de processamento de exames.

Código	Descrição
RN08	O status inicial do exame deve ser definido como “Pendente”.
RN09	Após a submissão pelo médico responsável, o exame é encaminhado ao módulo de IA e seu status é alterado para “Processando”.
RN10	Após a conclusão da inferência da IA, o status do exame deve ser alterado automaticamente para “Aguardando Revisão”, aguardando avaliação do médico responsável.
RN11	Apenas usuários com perfil médico podem registrar conclusões clínicas dos exames.
RN12	Exames “Concluído” ou “Com Divergência” não podem retornar automaticamente nem manualmente para o status de processamento.
RN13	Exames podem ser cancelados quando estiverem no status “Pendente”, e exames cancelados podem ser restaurados, retornando ao status “Pendente”.
RN14	Exames com falha no processamento podem ser restaurados, retornando ao status “Pendente” para nova submissão à análise por IA.
RN15	Apenas exames com status apropriado podem ser submetidos ao processamento automatizado.
RN16	Ao revisar o resultado da análise de IA, o médico deve informar se identificou divergência em relação à classificação automatizada, definindo o status final do exame como “Concluído” ou “Com Divergência”.
RN17	A impressão do exame em formato PDF só pode ser realizada quando o exame estiver nos status “Concluído” ou “Com Divergência”, contendo o resultado diagnóstico já registrado pelo médico responsável. A ação pode ser realizada tanto pelo MED quanto pelo GDC.

Fonte: Elaborado pela autora.

C.3 Análise de IA

Estas regras garantem a rastreabilidade e a integridade dos resultados gerados pelo módulo de IA, assegurando que cada inferência realizada permaneça devidamente documentada e vinculada ao exame correspondente. A Tabela 23 descreve as regras de negócio referentes à análise de exames por IA.

Tabela 23 – Regras de negócio de análise de IA.

Código	Descrição
RN18	Toda análise de IA deve permanecer vinculada ao exame correspondente.
RN19	Toda inferência realizada deve armazenar o modelo e a versão utilizados no processamento.
RN20	O sistema deve armazenar o nível de confiança associado à inferência realizada pela IA.
RN21	O sistema deve armazenar as imagens Grad-CAM geradas durante a análise automatizada.
RN22	Toda análise de IA deve possuir data e horário de execução registrados.

Fonte: Elaborado pela autora.

C.4 Segurança, Auditoria e Controle de Acesso

Estas regras estabelecem os mecanismos de segurança, rastreabilidade e controle de acesso do sistema, assegurando que operações críticas sejam registradas e que o acesso às funcionalidades seja restrito a usuários autorizados e ativos. A Tabela 24 descreve as regras de negócio referentes à segurança, auditoria e controle de acesso.

Tabela 24 – Regras de negócio de segurança, auditoria e controle de acesso.

Código	Descrição
RN23	Operações críticas realizadas no sistema devem gerar <i>logs</i> de auditoria.
RN24	Usuários inativos, ou vinculados a uma clínica inativa, não podem autenticar-se no sistema.
RN25	Usuários sem permissão adequada não podem acessar rotas protegidas do sistema.
RN26	O sistema deve manter histórico das alterações de status dos exames.
RN27	A exclusão de entidades clínicas e administrativas deve ocorrer preferencialmente por inativação lógica; para exames, o mecanismo equivalente é o cancelamento lógico.

Fonte: Elaborado pela autora.

APÊNDICE D – CASOS DE USO

Os UCs descrevem as principais interações entre os atores do sistema e as funcionalidades, derivados diretamente dos RFs levantados (Apêndice A). Cada caso de uso representa, de forma sintética, uma funcionalidade completa acionada por um ou mais atores, servindo de base para o diagrama apresentado na Subseção 5.2. A Tabela 25 descreve os UCs identificados, seus atores e os RFs correspondentes.

Tabela 25 – Casos de uso do sistema.

ID	Nome	Atores	Requisitos
UC01	Autenticação do usuário	ADM MED GDC	RF01–RF05
UC02	Editar próprio perfil	MED GDC	RF09
UC03a	Gerenciar usuários	ADM	RF06–RF13
UC03b	Gerenciar médicos da própria clínica	GDC	RF06–RF08, RF10–RF11
UC04	Gerenciar clínicas	ADM	RF14–RF19
UC05	Gerenciar dados da própria clínica	MED GDC	RF16
UC06	Gerenciar pacientes	ADM MED GDC	RF20–RF28
UC07	Submeter exame	MED	RF29–RF30, RF42
UC08a	Visualizar lista de exames	ADM MED GDC	RF31
UC08b	Consultar e baixar exames	MED	RF32, RF35, RF37
UC08c	Imprimir relatório do exame	MED GDC	RF41
UC09	Revisar resultado da análise de IA	MED	RF38–RF40
UC10	Analisar exame por IA	—	RF43–RF50
UC11	Consultar auditoria do sistema	ADM	RF51–RF55
UC12	Visualizar <i>dashboard</i>	ADM MED GDC	RF56–RF59

Fonte: Elaborado pela autora.

APÊNDICE E – CASOS DE TESTE

Os CTs descrevem a suíte automatizada implementada para verificar o funcionamento do sistema, organizados pelos mesmos módulos funcionais utilizados no Apêndice A, com indicação do tipo de teste (Subseção 2.5.10) e dos requisitos e RNs verificados.

E.1 Autenticação e Controle de Acesso

Estes testes verificam a autenticação de usuários, a validade e o versionamento de sessões por JWT, e a correta aplicação do RBAC em todas as rotas protegidas do sistema. A Tabela 26 descreve os CTs referentes à autenticação e ao controle de acesso.

Tabela 26 – Casos de teste de autenticação e controle de acesso.

ID	Tipo	Verificação	Requisito
CT01	Integração	API de autenticação, sessão e credenciais.	RF01–RF03
CT02	Unitário	Usuário ativo e versionamento de sessão JWT.	RF01, RF03
CT03	Integração	Matriz de perfis com todas as rotas protegidas.	RF04, RF05
CT04	Integração	Regressão da matriz de acesso por rota e perfil.	RF04
CT05	Unitário	Regressão do catálogo fechado de permissões.	RF04
CT06	Unitário	Fixação da matriz de permissões do ADM.	RF04

Fonte: Elaborado pela autora.

E.2 Gerenciamento de Usuários, Clínicas e Pacientes

Estes testes verificam as operações de cadastro, consulta e atualização de usuários, clínicas e pacientes, a inicialização e sincronização de papéis e permissões, e o isolamento de dados entre clínicas cadastradas. A Tabela 27 descreve os CTs referentes a esses módulos.

Tabela 27 – Casos de teste de usuários, clínicas e pacientes.

ID	Tipo	Verificação	Requisito
CT07	Integração	Invariantes e exposição de dados do módulo de usuários.	RF06–RF13
CT08	Integração	API e regras de negócio do módulo de clínicas.	RF14–RF19
CT09	Integração	Escopo de acesso, vínculos e transferência de pacientes.	RF20–RF28
CT10	Integração	Sincronização transacional da matriz de papéis e permissões.	RF13
CT11	Integração	Separação entre inicialização e reconciliação manual do RBAC.	RF13
CT12	Unitário	Regressão dos catálogos fechados de perfis e de <i>status</i> .	RF04, RN08
CT13	Unitário	Isolamento de dados entre clínicas, médicos e <i>logs</i> .	–

Fonte: Elaborado pela autora.

E.3 Gerenciamento de Exames

Estes testes verificam o ciclo de vida completo do exame, desde o cadastro e o *upload* da imagem até a conclusão da revisão médica e a impressão do relatório, incluindo o RBAC, o armazenamento seguro dos arquivos e o comportamento do fluxo diante de transições válidas e inválidas. A Tabela 28 descreve os CTs referentes ao gerenciamento de exames.

Tabela 28 – Casos de teste de gerenciamento de exames.

ID	Tipo	Verificação	Requisito
CT14	Integração	<i>Upload</i> , armazenamento, retenção e <i>download</i> por escopo.	RF30, RF35
CT15	Integração	Nomeação de arquivos e geração do pacote de imagens.	RF35
CT16	Integração	Controle de acesso ao histórico de exames.	RF37
CT17	Contrato	Mapeamento entre cada rota de exame e a permissão exigida.	RF31–RF41
CT18	Unitário	Autorização na camada de serviço.	RF31–RF41
CT19	Contrato	Formato de dados da listagem de exames.	RF31
CT20	Unitário	Persistência das imagens na hierarquia de arquivos.	RF29, RF30
CT21	Unitário	Resolução segura dos caminhos de arquivo dos exames.	RF29, RF30, RF35
CT22	Unitário	Tabela de transições de <i>status</i> , idempotência e revisão única.	RN08–RN16
CT23	Unitário	Máquina de estados sob concorrência real em PostgreSQL.	RN08–RN16
CT24	Unitário	Autorização de impressão do relatório do exame, restrita ao MED e ao GDC.	RF41, RN17
CT25	Unitário	Exclusividade do MED para revisar o resultado da análise.	RF38–RF40, RN11

Fonte: Elaborado pela autora.

E.4 Análise de Exames com IA

Estes testes verificam a integração entre o *backend* e o módulo de IA, incluindo o tratamento de falhas de comunicação, a persistência e o acesso aos mapas Grad-CAM gerados pela análise automatizada. A Tabela 29 descreve os CTs referentes à análise de exames com IA.

Tabela 29 – Casos de teste de análise de exames com IA.

ID	Tipo	Verificação	Requisito
CT26	Contrato	Comunicação entre <i>backend</i> e módulo de IA, incluindo erro, indisponibilidade e <i>timeout</i> .	RF42–RF47
CT27	Integração	<i>Download</i> do mapa Grad-CAM, sem exposição de caminhos.	RF46, RF47
CT28	Unitário	Persistência e resolução dos mapas de ativação Grad-CAM.	RF46

Fonte: Elaborado pela autora.

E.5 Auditoria e Rastreabilidade

Estes testes verificam se as operações críticas do sistema geram *logs* de auditoria completos, precisos e sem exposição de dados sensíveis, como senhas, *tokens* ou imagens de exames. A Tabela 30 descreve os CTs referentes à auditoria e à rastreabilidade.

Tabela 30 – Casos de teste de auditoria e rastreabilidade.

ID	Tipo	Verificação	Requisito
CT29	Unitário	Cobertura, privacidade (sem senhas/ <i>tokens</i> /imagens) e atomicidade dos <i>logs</i> de auditoria.	RF51, RN23
CT30	Unitário	Propagação segura de metadados de requisição (IP, <i>user-agent</i>) para a auditoria.	RF52

Fonte: Elaborado pela autora.

E.6 Infraestrutura e Persistência

Estes testes verificam requisitos não funcionais de persistência e validação de dados, como a consistência entre o esquema do banco de dados e as *migrations* aplicadas, e a rejeição de dados malformados enviados à API. A Tabela 31 descreve os CTs referentes à infraestrutura e à persistência.

Tabela 31 – Casos de teste de infraestrutura e persistência.

ID	Tipo	Verificação	Requisito
CT31	Contrato	Consistência estática entre o <i>schema</i> do banco e as <i>migrations</i> aplicadas.	RNF10, RNF12
CT32	Integração	Rejeição de campos desconhecidos nos corpos de requisição da API.	RNF05
CT33	Integração	Inicialização estrutural do banco e geração de massa de dados para ambiente acadêmico.	RNF09, RNF14

Fonte: Elaborado pela autora.

ANEXO
DECLARAÇÃO

Declaro para os devidos fins que este Trabalho de Conclusão de Curso (Monografia/Tese/Dissertação), escrito sob minha orientação, está em versão final, de acordo com as solicitações realizadas pela banca examinadora.

Informo também que procedi à revisão final do texto, constatando que atende às especificações das normas da ABNT para apresentação de trabalhos acadêmicos da UFCA, no que diz respeito ao conteúdo e à formatação.

Data e local da assinatura eletrônica

 Documento assinado digitalmente
LUANA BATISTA DA CRUZ
Data: 07/08/2026 09:45:34-0300
Verifique em <https://validar.iti.gov.br>