

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
INSTITUTO DE COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

VICTORIA BARROS DE ALMEIDA
YAGO DELFINO DA SILVA

SISCAR:

Um sistema para o gerenciamento da Clínica de Atendimento Referenciado

RIO DE JANEIRO
2025

VICTORIA BARROS DE ALMEIDA
YAGO DELFINO DA SILVA

SISCAR:

Um sistema para o gerenciamento da Clínica de Atendimento Referenciado

Trabalho de conclusão de curso de graduação
apresentado ao Instituto de Computação da
Universidade Federal do Rio de Janeiro como
parte dos requisitos para obtenção do grau de
Bacharel em Ciência da Computação.

Orientador: Profa. Juliana Baptista dos Santos França

RIO DE JANEIRO

2025

A447s

Almeida, Victoria Barros de
SISCAR: um sistema para o gerenciamento da Clínica de Atendimento
Referenciado / Victoria Barros de Almeida e Yago Delfino da Silva. – Rio de
Janeiro, 2026.

124 f.

Orientadora: Juliana Baptista dos Santos França.

Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação)-
Universidade Federal do Rio de Janeiro, Instituto de Computação, Bacharel em
Ciência da Computação, 2026.

1. Prontuário eletrônico. 2. Sistemas de informação em saúde. 3. Odontologia.
4. Desenvolvimento de software. 5. Clínica-escola. I. Silva, Yago Delfino da.
II. França, Juliana Baptista dos Santos (Orient.). III. Universidade Federal do Rio
de Janeiro, Instituto de Computação. IV. Título.

VICTORIA BARROS DE ALMEIDA
YAGO DELFINO DA SILVA


SISCAR:

Um sistema para o gerenciamento da Clínica de Atendimento Referenciado


Trabalho de conclusão de curso de graduação
apresentado ao Instituto de Computação da
Universidade Federal do Rio de Janeiro como
parte dos requisitos para obtenção do grau de
Bacharel em Ciência da Computação.

Aprovado em 12 de Março de 2026


BANCA EXAMINADORA:

Documento assinado digitalmente
 JULIANA BAPTISTA DOS SANTOS FRANÇA
Data: 18/03/2026 10:54:05-0300
Verifique em <https://validar.iti.gov.br>

Juliana Baptista dos Santos França
D.Sc. (UFRJ)

Documento assinado digitalmente
 ADRIANA SANTAROSA VIVACQUA
Data: 18/03/2026 11:33:31-0300
Verifique em <https://validar.iti.gov.br>

Adriana Vivacqua
D.Sc. (UFRJ)

Documento assinado digitalmente
 SILVANA ROSSETTO
Data: 18/03/2026 11:13:38-0300
Verifique em <https://validar.iti.gov.br>

Silvana Rossetto
D.Sc. (UFRJ)

Aos meus pais, Claudia e João Luiz, que com amor e dedicação me proporcionaram todas as oportunidades para chegar até aqui, e à minha segunda mãe, Cátia, que me transmitiu valores essenciais, que carrego com orgulho e gratidão.

Victoria Barros de Almeida

Dedico este trabalho à minha família, em especial à minha mãe Dalva Lucia Delfino, à minha prima Uyanã Neves e às minhas tias Angela Neves e Irene Neves, que sempre me apoiaram em qualquer situação e foram meu suporte e minha base desde a infância. Dedico também ao meu irmão Ygor Delfino, pois as inúmeras horas em que o via estudando e programando despertaram minha curiosidade e me levaram a cursar o Técnico em Informática e, posteriormente, a graduação em Ciência da Computação na UFRJ. Dedico, ainda, aos meus amigos que estiveram comigo nos melhores e piores momentos e tornaram minha vida e meu dia a dia mais leves: Bruno Ferrari, Mateus Vasconcelos, João Pedro Seda, Raphael Mesquita, Victoria Almeida, Prince, Pedro Henrique Zuben, Arthur Caccavo, Ana Clara Ceia, Francisco Moura e Raabe Vitoria.

Yago Delfino da Silva

AGRADECIMENTOS

Gostaríamos de expressar nossa sincera gratidão à Universidade Federal do Rio de Janeiro (UFRJ) pelo suporte institucional fornecido durante este período de estudo. Agradecemos aos departamentos, bibliotecários e demais funcionários que tornaram possível nossa jornada acadêmica, oferecendo recursos e assistência sempre que necessário.

Agradecemos, de forma especial, à nossa orientadora Prof.a Juliana Baptista dos Santos França pela orientação, confiança e apoio contínuos ao longo de toda a pesquisa. Também registramos nosso reconhecimento às professoras de Odontologia Rita de Cassia Ladeira (Universidade Federal Fluminense) e Maria do Céu Pinto do Amaral (Universidade Federal do Rio de Janeiro), cujas contribuições foram essenciais para a viabilização e o desenvolvimento deste projeto. Estendemos nossos agradecimentos a todos os funcionários da Clínica de Atendimento Referenciado (CAR), os professores, alunos e secretárias, pelo apoio e colaboração ao longo do trabalho.

"There is one thing that I can definitely say about the world as it exists right now. We are, at this moment, both creating and solving problems faster than we ever have before. So your job, the only thing anyone can ask of you as a human, is to solve more problems than you create."

Hank Green

RESUMO

Este trabalho apresenta o desenvolvimento do sistema SISCAR, um prontuário eletrônico voltado para a Clínica de Atendimento Referenciado (CAR) do Departamento de Odontologia Social e Preventiva da Universidade Federal do Rio de Janeiro (UFRJ), responsável pelo atendimento gratuito de alunos integrantes do programa de assistência estudantil. O principal objetivo do sistema é substituir os processos manuais baseados em papel, como o cadastro de pacientes, registro de prontuários clínicos, exames odontológicos e agendamento de consultas, por uma solução digital integrada, visando maior organização, segurança e eficiência no atendimento clínico. Para atingir esse objetivo, foi realizado um levantamento detalhado de requisitos junto aos principais stakeholders da clínica, adotando uma abordagem de desenvolvimento participativo e centrada no usuário. A partir desse levantamento, foram elaboradas modelagens conceituais, diagramas e protótipos de interface, seguidos pela implementação do sistema utilizando tecnologias modernas, como Flutter no frontend, Express.js no backend e PostgreSQL como banco de dados. O desenvolvimento foi orientado por testes, com o uso da metodologia TDD, além da aplicação de testes automatizados para validação das funcionalidades. Os resultados demonstram que o sistema atende às necessidades identificadas, permitindo acesso rápido ao histórico clínico dos pacientes, gestão estruturada das informações odontológicas e visualização organizada da agenda de consultas. Conclui-se que o SISCAR representa uma solução viável para a modernização dos processos da clínica, contribuindo para a melhoria do atendimento, o apoio ao ensino e a facilitação de análises e pesquisas acadêmicas.

Palavras-chave: prontuário eletrônico; sistemas de informação em saúde; odontologia; desenvolvimento de software; clínica-escola.

ABSTRACT

This work presents the development of the SISCAR system, an electronic health record designed for the Clínica de Atendimento Referenciado (CAR) of the Department of Social and Preventive Dentistry at the Federal University of Rio de Janeiro (UFRJ), which provides free dental care to students enrolled in the university assistance program. The main objective of the system is to replace manual, paper-based processes, such as patient registration, clinical records, dental examinations, and appointment scheduling, with an integrated digital solution, aiming to improve organization, security, and efficiency in both clinical and academic activities. To achieve this goal, a detailed requirements elicitation process was conducted with the clinic's main stakeholders, adopting a participatory and user-centered development approach. Based on the collected requirements, conceptual models, diagrams, and interface prototypes were created, followed by the system implementation using modern technologies, including Flutter for the frontend, Express.js for the backend, and PostgreSQL as the database management system. The development process was guided by test-driven development practices, supported by automated tests to ensure functional reliability. The results indicate that the system successfully meets the identified needs, enabling quick access to patients' clinical histories, structured management of dental information, and organized visualization of scheduled appointments. It is concluded that SISCAR represents a viable solution for modernizing the clinic's processes, contributing to improved patient care, academic support, and facilitation of data analysis and research activities.

Keywords: electronic health record; health information systems; dentistry; software development; teaching clinic.

LISTA DE ILUSTRAÇÕES

Figura 1 – Ciclo de vida de desenvolvimento	20
Figura 2 – Ciclo do TDD	22
Figura 3 – Modelagem Lógica do sistema SISCAR	29
Figura 4 – Modelagem Física do sistema SISCAR	33
Figura 5 – Protótipo das interfaces do sistema SISCAR desenvolvido no Figma	34
Figura 6 – Resultados dos testes automatizados e estrutura da suíte de testes	48
Figura 7 – Tela de <i>login</i> na versão <i>desktop</i> e <i>mobile</i>	52
Figura 8 – Tela inicial na versão <i>desktop</i>	53
Figura 9 – Tela inicial na versão <i>mobile</i>	54
Figura 10 – Tela de agendamento na versão <i>desktop</i> e <i>mobile</i>	55
Figura 11 – Tela de edição de agendamento na versão <i>desktop</i> e <i>mobile</i>	56
Figura 12 – Tela da lista de professores na versão <i>desktop</i> e <i>mobile</i>	57
Figura 13 – Tela de cadastro de professor na versão <i>desktop</i> e <i>mobile</i>	59
Figura 14 – Tela de edição de professor na versão <i>desktop</i> e <i>mobile</i>	61
Figura 15 – Tela da lista de alunos na versão <i>desktop</i> e <i>mobile</i>	62
Figura 16 – Tela de cadastro de aluno na versão <i>desktop</i> e <i>mobile</i>	64
Figura 17 – Tela de edição de aluno na versão <i>desktop</i> e <i>mobile</i>	65
Figura 18 – Tela da lista de secretárias na versão <i>desktop</i> e <i>mobile</i>	66
Figura 19 – Tela de cadastro de secretária na versão <i>desktop</i> e <i>mobile</i>	67
Figura 20 – Tela de edição de secretária na versão <i>desktop</i> e <i>mobile</i>	68
Figura 21 – Tela da lista de pacientes na versão <i>desktop</i> e <i>mobile</i>	69
Figura 22 – Tela de <i>menu</i> de um paciente no <i>mobile</i>	70
Figura 23 – Tela de dados pessoais de paciente na versão <i>desktop</i> e <i>mobile</i>	72
Figura 24 – Tela preenchida de dados pessoais de paciente na versão <i>desktop</i> e <i>mobile</i>	74
Figura 25 – Tela de anamnese de paciente na versão <i>desktop</i> e <i>mobile</i>	75
Figura 26 – Tela de lista de anamnese na versão <i>desktop</i> e <i>mobile</i>	77
Figura 27 – Tela de odontograma de paciente na versão <i>desktop</i>	78
Figura 28 – <i>Modal</i> do dente 21 no odontograma	79
Figura 29 – Tela de odontograma de paciente na versão <i>mobile</i>	80
Figura 30 – Tela de lista de odontograma na versão <i>desktop</i> e <i>mobile</i>	81
Figura 31 – Tela exames de biofilme de paciente na versão <i>desktop</i> e <i>mobile</i>	83
Figura 32 – <i>Modal</i> de Fatores Retentivos	85
Figura 33 – Tela da lista de exames biofilme na versão <i>desktop</i> e <i>mobile</i>	86
Figura 34 – Tela exames de sangramento de paciente na versão <i>desktop</i> e <i>mobile</i>	88
Figura 35 – Tela da lista de exames sangramento na versão <i>desktop</i> e <i>mobile</i>	90

Figura 36 – Tela exames cpos de paciente na versão <i>desktop</i> e <i>mobile</i>	92
Figura 37 – <i>Modal</i> do dente 14 no exame cpos	93
Figura 38 – Tela da lista de exames cpos na versão <i>desktop</i> e <i>mobile</i>	94

LISTA DE QUADROS

Quadro 1 – Dicionário de dados de cadastro	24
Quadro 2 – Requisitos Funcionais do Sistema SISCAR	25
Quadro 3 – Requisitos Não Funcionais do Sistema SISCAR	27

SUMÁRIO

1	INTRODUÇÃO	15
1.1	PROBLEMA	15
1.2	OBJETIVO	16
1.3	JUSTIFICATIVA	17
2	REFERENCAL TEÓRICO	18
2.1	CONCEITOS	18
2.1.1	Design Participativo	18
2.1.2	Experiência de Usuário (<i>UX</i>)	19
2.1.3	Métodos Ágeis	20
2.1.4	Test Driven Development (<i>TDD</i>) e <i>Minimum Viable Product</i> (<i>MVP</i>)	21
2.1.5	Design Patterns	22
3	DESENVOLVIMENTO DO SOFTWARE	24
3.1	MODELAGEM E PROTOTIPAGEM	24
3.1.1	Levantamento de Requisitos	24
3.1.1.1	Requisitos Funcionais	24
3.1.1.2	Requisitos Não-Funcionais	27
3.1.2	Fluxo do Usuário	28
3.1.3	Modelagem Lógica	28
3.1.4	Prototipagem de Interfaces com Figma	34
3.2	ARQUITETURA DA SOLUÇÃO PROPOSTA	34
3.2.1	Componentes Principais e Inter-relações	34
3.2.2	Arquitetura de Sistema e Instanciamento	35
3.2.3	Tecnologias de Implementação	35
3.3	FRONTEND	36
3.3.1	O porquê do uso do Flutter	36
3.3.2	Arquitetura e Estrutura de Pastas	36
3.3.3	Responsividade	38
3.3.4	Temas e Tokens	39
3.3.5	Tratamento de Erros	40
3.3.5.1	Tipagem e Semântica de Exceções	40
3.3.5.2	Normalização na Camada de Serviço	40
3.3.5.3	Camada de Repositório: Contextualização	40
3.3.5.4	Camada de ViewModel: Gestão de Estado	41

3.3.5.5	Camada de View: Apresentação ao Usuário	41
3.3.6	Persistência e sessão	42
3.3.6.1	Infraestrutura de Armazenamento	42
3.3.6.2	Ciclo de Vida da Sessão	42
3.3.6.3	Segurança e Robustez	43
3.3.7	Injeção de Dependências e Gerenciamento de Escopos	43
3.4	BACKEND	44
3.4.1	O porquê do uso do Node.js com Express	44
3.4.2	Arquitetura e Estrutura de Pastas	44
3.4.3	Segurança e Controle de Acesso	45
3.4.4	Integração com o banco de dados PostgreSQL	46
3.5	DOCKER E CONTAINERIZAÇÃO	47
3.6	TESTES	48
3.7	VERIFICAÇÃO DE REQUISITOS E AVALIAÇÃO COM USUÁRIOS	49
4	RESULTADOS	51
4.1	TELAS DO APLICATIVO	52
4.1.1	Tela de Login	52
4.1.2	Tela Inicial	53
4.1.3	Tela de Agendamento	55
4.1.4	Tela de Professores – Listagem	57
4.1.5	Tela de Professores – Cadastro e Edição	59
4.1.6	Tela de Alunos – Listagem	62
4.1.7	Tela de Alunos – Cadastro e Edição	64
4.1.8	Tela de Secretárias – Listagem	66
4.1.9	Tela de Secretárias – Cadastro e Edição	67
4.1.10	Tela de Pacientes – Listagem	69
4.1.11	Tela de Paciente – Menu	70
4.1.12	Tela de Paciente – Dados Pessoais	72
4.1.13	Tela de Paciente – Anamnese	75
4.1.14	Tela de Paciente – Odontograma	78
4.1.15	Tela de Paciente – Exame de Biofilme	83
4.1.16	Tela de Paciente – Exame de Sangramento	88
4.1.17	Tela de Paciente – Exame de CPOS	92
4.2	TRABALHOS RELACIONADOS	95
4.2.1	Dental Office	95
4.2.2	Dental Escola	96
4.2.3	Diferenças e semelhanças em relação ao SISCAR	97
5	CONCLUSÃO	99

5.1	LIMITAÇÕES	100
5.2	TRABALHOS FUTUROS	100
	REFERÊNCIAS	102
	ANEXO A – FICHA DO PACIENTE UTILIZADA NA CLÍNICA DE ATENDIMENTO REFERENCIADO	104
	ANEXO B – CONTRATO ENTRE A CAR E A SUPEREST (PR-7)	117

1 INTRODUÇÃO

A junção do universo computacional e odontológico à primeira vista pode parecer recente, mas na realidade essa combinação tem raízes que se estendem por décadas. Em 1950 Robert Ledley, médico dentista de Nova York, foi pioneiro ao usar um dos poucos computadores existentes no mundo, na *National Bureau Standards* (NBS), para analisar as características de determinados pacientes, para projetos na área da Medicina Dentária (LEDLEY, 2022). Desde então, o uso de tecnologia da informação ampliou-se de forma drástica na área da saúde como um todo e em particular na Odontologia (HEID JOSEPH E. CHASTEEN, 2002).

Em 2005, o Centro de Informática Odontológica da Faculdade de Medicina Dentária da Universidade de Pittsburgh realizou uma pesquisa em uma amostra aleatória de dentistas gerais nos Estados Unidos e descobriu que 25% deles usavam um computador no consultório e 1,8% não usavam papel (isto é, todos os registros dos pacientes eram mantidos eletronicamente, sem *backup* em *paper*) (SCHLEYER, 2001).

Neste mesmo ano, no início de setembro, a Clínica de Atendimento Referenciado (CAR) foi inaugurada, com o objetivo de atender servidores da UFRJ. Após a universidade adotar o sistema de cotas raciais em 2014 (SILVA, 2022), a gerência da clínica entrou em contato com a SuperEst (Superintendência Geral de Políticas Estudantis) para, também, atender alunos de ações afirmativas. Atualmente esse diálogo é feito com a PR7 (Pró-reitoria de Políticas Estudantis); esta entidade é responsável pela inscrição dos alunos cotistas, realizando um sorteio para selecionar uma lista de alunos que serão encaminhados à clínica para dar início ao atendimento/tratamento (PR7, 2022).

Desde então no mesmo escopo da pesquisa anterior, houve um aumento drástico, superior ao dobro, na informatização das clínicas odontológicas (BENOIT BUKIET FRÉ-DÉRIC, 2022), porém a CAR continua realizando todos os seus processos manualmente, apesar desses terem se tornado mais complexos com o passar do tempo.

1.1 PROBLEMA

Clínicas odontológicas requerem uma gestão precisa e eficiente, tanto das operações clínicas quanto administrativas. Neste trabalho, porém, existe uma perspectiva específica de uma clínica inserida em uma faculdade de odontologia de uma universidade pública que integra suas atividades com a formação profissional de seus estudantes. Este contexto traz um grande desafio, manter e equilibrar uma combinação complexa de atividades, incluindo cuidados clínicos, educação, pesquisa e administração.

A clínica odontológica, desde o início de suas atividades, adota o cadastro de seus pacientes de forma manual. Já o agendamento é realizado em uma comunicação tele-

fônica com o paciente e, posteriormente, registrado em uma agenda física por uma das secretárias. Embora este método tenha sido eficaz por um tempo, com o crescimento e evolução da clínica percebem-se algumas problemáticas como, uma lentidão no trabalho, perda de dados e principalmente confusões por dados inconsistentes. Como a clínica é gerenciada por mais de um professor, nota-se uma grande falta de sincronização na hora do preenchimento de dados, levando professores e alunos se confundirem acerca de que procedimentos foram e devem ser realizados.

Além disso, uma outra dificuldade comum enfrentada pela clínica é a localização rápida das fichas dos pacientes em meio a uma grande quantidade de documentos físicos. Isso não apenas consome tempo das secretárias e clínicos, mas também pode levar a atrasos no atendimento, erros de registro e até mesmo a perda irreparável de informações importantes sobre o histórico de tratamento e condições médicas dos pacientes.

Ademais, a grande quantidade de documentos físicos também dificulta a realização de pesquisa e análise de dados da clínica. Essas análises são essenciais não apenas para embasar decisões futuras, como determinar a necessidade de reabastecer o estoque, mas também no apoio às atividades de ensino e pesquisa. Visto que a clínica está integrada à uma universidade pública é de suma importância que os dados possam ser prontamente acessados e analisados para a produção destes artigos e outras contribuições científicas. Sendo assim, tornou-se evidente a necessidade de modernização e otimização de processos.

No intuito de atender às necessidades citadas, este trabalho propõe a criação de um sistema que simplifique a gestão e atendimentos da clínica.

1.2 OBJETIVO

O objetivo geral deste trabalho é automatizar os processos de gestão da clínica dentária, até então exercidos manualmente. Para isso, foi desenvolvido um sistema multiplataforma (*web* e *mobile*), permitindo que a clínica usufrua de diversas funcionalidades, como agendamento de consultas, acompanhamento de históricos odontológicos e gestão de pacientes, com o intuito de melhorar a eficiência e o funcionamento da CAR.

Constituem os objetivos secundários deste trabalho:

- Modelar os requisitos: o primeiro passo é criar uma estrutura funcional e prática para o sistema, identificando os principais processos e operações diárias que a clínica realiza.
- Desenvolver e implementar o sistema multiplataforma: o sistema deve ser desenvolvido com uma interface amigável ao usuário, exercendo as funcionalidades básicas que facilitem a gestão e atendimento da clínica.

- Ensinar e instruir o uso do sistema: é fundamental a realização de testes e a formulação de tutoriais de uso, principalmente porque os clínicos podem não estar habituados a usar sistemas informatizados em sua rotina de trabalho.

1.3 JUSTIFICATIVA

O estímulo para o desenvolvimento deste trabalho originou-se da experiência de um dos autores do projeto, que foi selecionado para o atendimento na clínica. Durante uma das consultas, o autor constatou uma dificuldade significativa relatada pela professora responsável pela clínica, relacionada à dependência dos métodos tradicionais e manuais de gestão. A professora destacou como a quantidade de fichas de pacientes e a dificuldade em localizá-las rapidamente impactavam a eficiência e a qualidade dos serviços prestados. Além disso, a confusão no agendamento dos alunos foi identificada como um problema crucial. Esse problema tornou-se ainda mais evidente quando, após o agendamento inicial do autor, ele foi marcado para comparecer à clínica depois de ter recebido alta do tratamento anterior.

Com base em pesquisas na área (SILVEIRA et al., 2006), a informatização é cada vez mais comum entre os profissionais da odontologia e os dados levantados indicam que a grande maioria dos entrevistados concorda que a presença de um computador (e consequentemente a digitalização de informações) auxilia no atendimento e no gerenciamento de clínicas.

Em particular percebeu-se a oportunidade de aplicar os conhecimentos adquiridos na disciplina eletiva, Software Livre e Metodologias Participativas, administrada pelo professor Celso Avelar, cursada por ambos os autores. Nela se entende como papel crucial da universidade analisar e tratar problemas reais, sem agentes e cenários hipotéticos. Assim, foi usado conceitos ensinados na disciplina como Tecnologia Social, Métodos Ágeis e Desenvolvimento Participativo.

2 REFERENCAL TEÓRICO

Este capítulo tem como objetivo apresentar os fundamentos teóricos que sustentam as decisões metodológicas e técnicas adotadas no desenvolvimento deste trabalho. Para isso, ele está organizado em seções que abordam, de forma progressiva, os principais conceitos utilizados no projeto: *Design* Participativo e Experiência do Usuário (*UX*), métodos ágeis com foco em *Scrum*, práticas de *Test Driven Development (TDD)* e *Minimum Viable Product (MVP)*, além de padrões de projeto (*Design Patterns*). Essa estrutura busca estabelecer a base conceitual que orienta a análise, a implementação e a avaliação da solução proposta.

2.1 CONCEITOS

2.1.1 Design Participativo

Foi adotado o método democrático de pesquisa-ação conhecido como *Design* Participativo (DP). Sua origem remonta aos países escandinavos na década de 1970, em um contexto de intensos movimentos políticos, nos quais as pessoas passaram a reivindicar maior participação nas decisões que impactavam suas vidas (IBARRA, 2021). Durante esse período, o *Design* Participativo se destacou como uma ferramenta crucial para o desenvolvimento de tecnologias, pois tinha como objetivo democratizar o processo de criação de tecnologias de informação e comunicação permitindo que os próprios usuários influenciassem as decisões sobre o *design* das ferramentas, programas e sistemas que iriam utilizar, garantindo que suas necessidades e perspectivas fossem incorporadas desde as primeiras etapas do processo.

Esse método foi utilizado, por exemplo, para capacitar sindicatos, com o objetivo de entender como os trabalhadores poderiam se beneficiar de inovações em TI, que pretendiam melhorar o ambiente de trabalho. Assim, o DP se mostrou um meio eficaz para envolver diretamente os usuários na criação de soluções que atendem suas necessidades, promovendo um ambiente mais justo e colaborativo no contexto do trabalho (BJÖGVINSSON PELLE EHN, 2012).

Embora o termo "*design*" seja amplamente utilizado no sentido comum, especialmente em inglês, para se referir à estética ou ao visual de algo, como na camada de interface (*Front-End*), o conceito de "*design*" vai muito além dessa interpretação superficial. No contexto da computação, e mais especificamente em áreas como o desenvolvimento de sistemas, o "*design*" deve ser entendido de forma mais ampla, como o plano de construção de um projeto. Neste sentido, "*design*" envolve o planejamento e a estruturação de todo o processo de criação de uma solução, englobando as partes interessadas (*stakeholders*) de várias maneiras em vários estágios do trabalho.

Sendo assim, entende-se DP como colocar a participação dos usuários do sistema no centro das preocupações do desenvolvimento do mesmo como um todo, não apenas em decisões estéticas mas também em toda a etapa de desenvolvimento que lhe cabe. Já que os usuários são especialistas em seu ambiente de trabalho e muitas vezes tem uma visão única de suas falhas e potencial. Com “participação” por sua vez entende-se, em um contexto computacional, como diálogo e troca de conhecimentos entre os desenvolvedores e o usuário não apenas de forma consultiva mas promovendo aprendizagem mútua entre todos os participantes durante o processo. (MULLER, 2007).

Essa abordagem tem se mostrado eficaz para alcançar uma boa experiência de uso dos sistemas, por, potencialmente melhor contemplar preferências e necessidades. (AMSTEL, 2009) (CHERRY, 1999) .

2.1.2 Experiência de Usuário (UX)

De acordo com (HARTSON; PYLA, 2018), a Experiência de Usuário (UX), ou *User Experience*, refere-se a todo o conjunto de sensações, emoções e percepções de uma pessoa ao interagir com um produto, sistema ou serviço. Vai muito além da estética, engloba como o usuário se sente antes, durante e depois dessa interação. No contexto do desenvolvimento de *software*, uma boa experiência de usuário significa que o sistema não apenas cumpre suas funcionalidades, mas o faz de uma maneira que seja intuitiva, eficiente, agradável e significativa para quem o utiliza.

A busca por uma UX positiva é fundamental para o sucesso de qualquer sistema, pois impacta diretamente a satisfação do usuário, a produtividade e a adoção da ferramenta. Um sistema com má UX pode levar à frustração, erros frequentes e, em última instância, ao abandono, como alertam (KRUG, 2000) ao discutir a importância da clareza e simplicidade no *design*.

Considerando que nosso projeto adota o *Design Participativo*, onde os usuários são colocados no centro do processo de desenvolvimento, a Experiência de Usuário emerge como o resultado direto e o principal indicador de sucesso dessa abordagem colaborativa. Ao envolver os usuários ativamente, é possível incorporar suas necessidades, expectativas e conhecimentos práticos, garantindo que o sistema seja não apenas funcional, mas também genuinamente útil e agradável para quem irá utilizá-lo no dia a dia.

Essa ligação entre o *Design Participativo* e a Experiência do Usuário é evidenciada em projetos práticos, como o *redesign* do *website* da Biblioteca da Universidade de Illinois em Chicago (UIC). Neste caso, a adoção de métodos participativos, como o *Object-Oriented UX (OOUX)* e a inclusão de *stakeholders* nos testes de usabilidade, garantiu que o projeto fosse entregue dentro do prazo, assegurando que as decisões de *design* fossem baseadas em pesquisas reais com os usuários finais em vez de especulações (WOOD; KOMPARE, 2017). Tais exemplos demonstram que o DP atua como uma poderosa metodologia de *UX Research*, elevando a credibilidade e a eficácia das decisões de *design*.

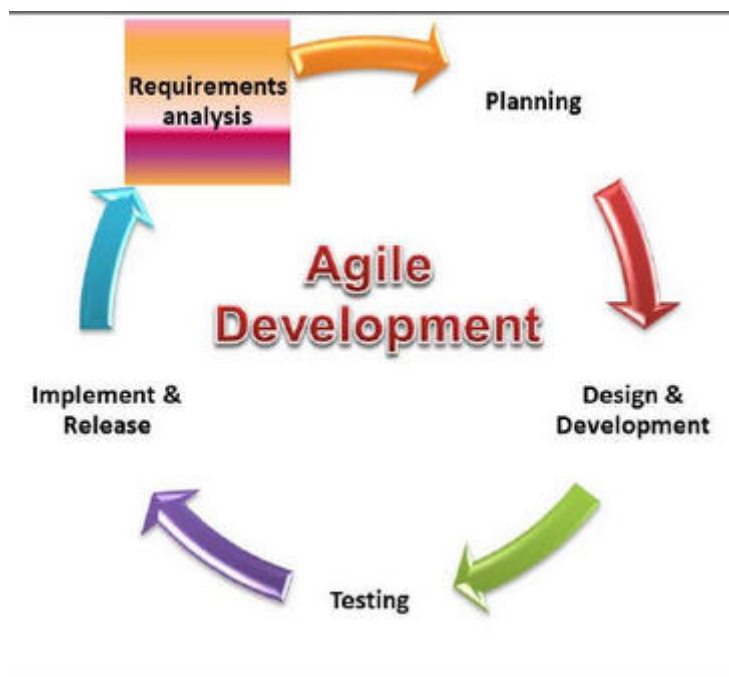
Além disso, no contexto de um desenvolvimento guiado por metodologias ágeis, a Experiência de Usuário é aprimorada por meio de iterações curtas e frequentes. As iterações do desenvolvimento ágil permitem que o *design* de UX seja continuamente validado e ajustado. Essa integração contínua entre *design* e desenvolvimento ágil assegura que o produto final não só seja funcional, mas que também proporcione uma interação otimizada e satisfatória (MORENO; ALHAMMAD, 2022).

2.1.3 Métodos Ágeis

É fundamental que o desenvolvimento de um sistema conduzido por uma dupla a partir de um TCC (onde o tempo de desenvolvimento é limitado), seja auto-organizado e colaborativo. Essa aspiração, também se reflete no “Manifesto Ágil” (BECK et al., 2001), e duas décadas após sua publicação as metodologias ágeis ganharam ampla aceitação (DINGSOYRA et al., 2012). Seus ideais de soluções flexível concentradas nas necessidades do “cliente”, também se assemelham aos do *Design* Participativo, que já foi abordado, sendo cruciais para incorporar as perspectivas dos usuários desde as primeiras etapas do projeto. Dessa forma, a abordagem ágil se torna um pilar essencial para garantir a adaptabilidade e a resposta eficaz às demandas em constante evolução do nosso projeto.

O ciclo de vida de desenvolvimento de *software* (SDLC) ágil integra planejamento, desenvolvimento, e testes em um processo cíclico, iterativo como pode-se ver na imagem a seguir.

Figura 1 – Ciclo de vida de desenvolvimento



Fonte: *Analyzing Agile Development – from Waterfall Style to Scrumban*

Para as iterações foi usado o *Scrum*, um subconjunto de Métodos Ágeis, que refina ainda mais esses princípios em uma *framework* estruturada, tornando-a uma das metodologias ágeis mais populares. Não serão adotados todos os artefatos do *framework Scrum*, já que por exemplo um *Scrum Master* em uma equipe de apenas duas pessoas não faria sentido. Porém, conceito de *Backlog* (listar e priorizar todas as funcionalidades necessárias para o sistema) e *Sprint* (ciclos de desenvolvimento com duração fixa, geralmente de uma a quatro semanas), serão muito úteis para a organicidade do processo. Esses elementos nos permitem manter o foco, gerenciar o progresso de forma iterativa e obter *feedback* constante, elementos cruciais para o desenvolvimento de um Produto Mínimo Viável (*MVP*) e a implementação de *Test Driven Development* (*TDD*), que serão detalhados a seguir.

2.1.4 Test Driven Development (*TDD*) e *Minimum Viable Product* (*MVP*)

TDD é a sigla para *Test Driven Development* (Desenvolvimento Dirigido por Testes), essa ferramenta ou método de desenvolvimento, tem como objetivo fazer com que cada desenvolvedor escreva seu código de teste, antes do código de produção.

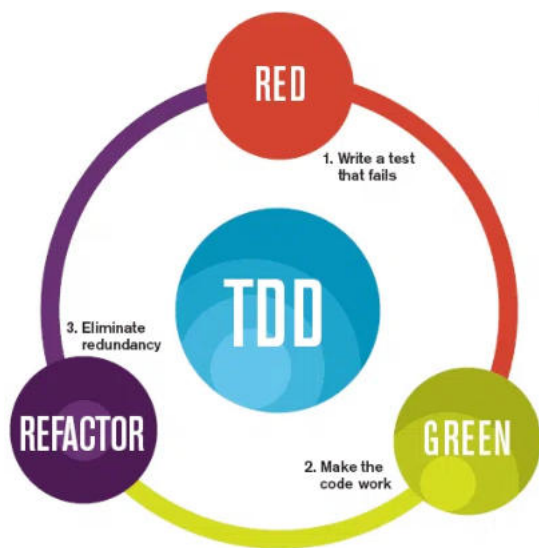
Assim como em métodos ágeis, *TDD* consiste em pequenas iterações de desenvolvimento de testes unitários para cada funcionalidade a ser implementada. E como o desenvolvimento ágil demanda *feedbacks* regulares para o desenvolvimento de um produto de qualidade, utilizar *TDD* se torna quase essencial para o desenvolvimento do nosso sistema.

Em muitos casos, o volume de testes que os desenvolvedores escrevem tende a ser maior do que o código de produção efetivo. Essa prática é intencional: os desenvolvedores são levados a criar testes unitários precisos que mapeiam o comportamento esperado de cada funcionalidade. A necessidade desse processo, em que os testes falham inicialmente para depois guiar a implementação, reside no valor que os testes oferecem como ferramenta de *design* e especificação.

O processo de desenvolvimento seguiu o ciclo de três etapas conhecido como *red-green-refactor*, que é fundamental para a qualidade do código.

A adoção do *TDD* e suas práticas está diretamente ligada à construção do Mínimo Produto Viável (*MVP*). O *MVP* é uma versão inicial de um produto em processo de desenvolvimento que já possui funcionalidades suficientes para ser testado e obter *feedback* em futuras iterações. A parte “mínima” do *MVP* é especialmente crítica, pois o objetivo não é entregar um produto completo, e sim um produto que permita testar os conceitos básicos sob as necessidades do usuário para validar o rumo que o desenvolvimento está tomando.

Figura 2 – Ciclo do TDD



Fonte: dev.to

- **Red:** Escrever um teste que falha, pois a funcionalidade ainda não foi implementada.
- **Green:** Implementar o código mínimo necessário para que o teste seja executado com sucesso.
- **Refactor:** Após os testes passarem, reorganizar o código para melhorar sua legibilidade, manutenibilidade e *design*, sem alterar sua funcionalidade.

2.1.5 Design Patterns

O termo *Design Pattern* (Padrão de Projeto) refere-se a soluções reutilizáveis e comprovadamente eficazes para problemas recorrentes encontrados durante o desenvolvimento de *software* (GAMMA et al., 1994). Esses padrões não consistem em blocos de código prontos, mas em descrições de estruturas e interações entre classes e objetos que podem ser adaptadas a diferentes contextos, promovendo flexibilidade, legibilidade e facilidade de manutenção do sistema.

O conceito de padrões de projeto não surgiu originalmente no campo da computação. Suas raízes remontam ao trabalho do arquiteto Christopher Alexander, que, em sua obra *A Pattern Language: Towns, Buildings, Construction* (1977), descreveu um “vocabulário” para o *design* de ambientes urbanos e arquitetônicos, no qual cada padrão representava uma solução recorrente para problemas de construção e organização espacial. Essa ideia de capturar soluções eficazes em forma de “linguagem de padrões” foi posteriormente adaptada à engenharia de *software* por Erich Gamma, Richard Helm, Ralph Johnson e John Vlissides. Em 1994, os autores publicaram o livro *Design Patterns: Elements of Reusable Object-Oriented Software*, no qual aplicaram o conceito de padrões ao desenvolvimento orientado a objetos, apresentando 23 soluções clássicas para problemas recorrentes de *design*. A obra rapidamente se tornou referência mundial, sendo popularmente conhecida como o “livro da Gang of Four” (*GoF*) (GAMMA et al., 1994). Desde então, novos padrões têm sido identificados e aplicados em diversos domínios da computação, consolidando o uso da abordagem de padrões como uma prática essencial no *design* de *software* (SHVETS, 2014).

Entre os principais benefícios do uso de padrões de projeto, destacam-se:

- **Reutilização de soluções:** evita a reinvenção de soluções para problemas recorrentes;
- **Facilidade de manutenção e evolução:** promove clareza estrutural e reduz o acoplamento entre componentes;
- **Comunicação entre desenvolvedores:** estabelece uma linguagem comum entre profissionais, facilitando a compreensão e a colaboração em equipe;
- **Aderência a boas práticas de engenharia:** favorece a criação de sistemas mais robustos, modulares e escaláveis.

Os padrões de projeto podem ser classificados em três grandes categorias:

1. **Padrões Criacionais:** tratam da forma como os objetos são instanciados, promovendo flexibilidade na criação e no gerenciamento de dependências (ex.: *Singleton*, *Factory Method*, *Builder*);
2. **Padrões Estruturais:** lidam com a composição de classes e objetos, otimizando a organização e a relação entre eles (ex.: *Adapter*, *Composite*, *Facade*);
3. **Padrões Comportamentais:** tratam da comunicação entre objetos e da distribuição de responsabilidades (ex.: *Observer*, *Strategy*, *Command*).

A adoção desses padrões contribuiu para o desenvolvimento de um sistema mais sustentável e escalável, assegurando qualidade técnica e coerência arquitetural, em consonância com as metodologias ágeis e o *Design Participativo* adotados neste trabalho.

3 DESENVOLVIMENTO DO SOFTWARE

3.1 MODELAGEM E PROTOTIPAGEM

3.1.1 Levantamento de Requisitos

O levantamento de requisitos do sistema SISCAR foi realizado com os principais *stakeholders* envolvidos na clínica, em reuniões que ocorreram tanto de forma presencial quanto *online*. O levantamento de requisitos consiste na identificação das necessidades de usuário e das restrições que o sistema deve operar, sendo uma etapa fundamental no desenvolvimento de *software*.

Além das reuniões, foi realizada a análise do prontuário utilizada na clínica, a qual serviu como base para a definição das informações essenciais a serem registradas no sistema. Esse documento representa o principal instrumento de registro das informações dos pacientes na clínica, concentrando dados pessoais, anamnese e exames.

Esse processo orientou a modelagem lógica e física do banco de dados e a definição das funcionalidades do sistema, garantindo que o prontuário eletrônico refletisse fielmente a realidade da clínica, ao mesmo tempo em que promovesse maior organização, segurança e rastreabilidade das informações.

Além disso, a análise e as reuniões evidenciaram a necessidade de controle de acesso aos dados, considerando os diferentes perfis de usuários envolvidos (alunos, professores e secretaria), bem como o registro detalhado das ações realizadas no sistema, como quem realizou determinado procedimento, em qual data. Esses requisitos reforçam o papel do sistema não apenas como ferramenta operacional, mas também como apoio ao processo de aprendizagem e à produção acadêmica.

3.1.1.1 Requisitos Funcionais

Para esclarecer os requisitos de cadastro, o Quadro 1 apresenta o dicionário de dados com os campos registrados em cada tipo de cadastro.

Quadro 1 – Dicionário de dados de cadastro

Cadastro	Dados registrados
Aluno, Professor e Secretária	Nome Civil, Nome Social, CPF, E-mail, Senha e Confirmar Senha.
Paciente	Nome Civil, Nome Social, CPF, Celular, E-mail, Estado, Cidade, CEP, Bairro, Rua, Número, Complemento, Data de Nascimento, Raça e Gênero.

Quadro 2 – Requisitos Funcionais do Sistema SISCAR

Código	Nome	Descrição	Prioridade	Papel
RF01	Gerenciar perfil	Através do <i>login</i> , o sistema deverá identificar quais funcionalidades o usuário possui permissão de acesso.	Essencial	Professor
RF02	Cadastrar aluno	O sistema deve permitir o cadastro de aluno, registrando os dados definidos no dicionário de cadastro.	Essencial	Professor
RF03	Editar aluno	O sistema deve permitir a alteração das informações dos alunos cadastrados.	Importante	Professor
RF04	Deletar aluno	O sistema deve permitir a remoção de alunos cadastrados.	Importante	Professor
RF05	Visualizar aluno	O sistema deve permitir a visualização das informações do aluno.	Importante	Professor
RF06	Cadastrar paciente	O sistema deve permitir o cadastro de pacientes, registrando os dados definidos no dicionário de cadastro.	Essencial	Aluno e Professor
RF07	Deletar paciente	O sistema deve permitir a remoção de pacientes cadastrados.	Essencial	Professor
RF08	Editar paciente	O sistema deve permitir a alteração das informações dos pacientes cadastrados.	Essencial	Aluno e Professor
RF09	Visualizar paciente	O sistema deve permitir a visualização das informações do paciente.	Importante	Aluno e Professor
RF10	Cadastrar secretária	O sistema deve permitir o cadastro de secretária, registrando os dados definidos no dicionário de cadastro.	Essencial	Professor
RF11	Visualizar secretária	O sistema deve permitir a visualização das informações do secretária.	Importante	Professor
RF12	Deletar secretária	O sistema deve permitir a remoção de secretária cadastrados.	Essencial	Professor

Código	Nome	Descrição	Prioridade	Papel
RF13	Editar secretária	O sistema deve permitir a alteração das informações dos secretária cadastrados.	Essencial	Professor
RF14	Visualizar secretária	O sistema deve permitir a visualização das informações dos secretária.	Importante	Professor
RF15	Cadastrar professor	O sistema deve permitir o cadastro de professor, registrando os dados definidos no dicionário de cadastro.	Essencial	Professor
RF16	Deletar professor	O sistema deve permitir a remoção de professores cadastrados.	Essencial	Professor
RF17	Editar professor	O sistema deve permitir a alteração das informações dos professores cadastrados.	Essencial	Professor
RF18	Visualizar professor	O sistema deve permitir a visualização das informações dos professores.	Importante	Professor
RF19	Cadastrar prontuário	O sistema deve permitir o cadastro de informações da consulta, vinculando-as a um paciente.	Essencial	Aluno e Professor
RF20	Deletar prontuário	O sistema deve permitir a remoção de prontuários cadastrados.	Essencial	Professor
RF21	Editar prontuário	O sistema deve permitir a alteração das informações dos prontuários cadastrados.	Essencial	Aluno e Professor
RF22	Visualizar prontuário	O sistema deve permitir a visualização das informações dos prontuários.	Importante	Aluno e Professor
RF23	Agendar consultas	O sistema deve permitir o agendamento de consultas no calendário, vinculando professor e paciente.	Essencial	Secretária e Professor
RF24	Cancelar consultas	O sistema deve permitir o cancelamento de consultas agendadas.	Essencial	Secretária e Professor
RF25	Editar consultas	O sistema deve permitir a alteração das consultas agendadas.	Essencial	Secretária e Professor

3.1.1.2 Requisitos Não-Funcionais

Quadro 3 – Requisitos Não Funcionais do Sistema SIS-CAR

Código	Nome	Descrição	Categoria
RNF01	Segurança no acesso	O sistema deve garantir que apenas pessoas autorizadas tenham acesso às informações armazenadas.	Confiabilidade
RNF02	Usabilidade	Os usuários devem ser capazes de operar o sistema após um período de treinamento adequado.	Usabilidade
RNF03	Portabilidade	O sistema deve ser capaz de funcionar em diferentes plataformas e navegadores.	Usabilidade
RNF04	Proteção de dados	O sistema deve proteger as informações armazenadas contra acessos não autorizados, sejam eles humanos ou automatizados.	Confiabilidade
RNF05	Nível de acesso	O sistema deve garantir que cada usuário tenha acesso apenas às funcionalidades e informações permitidas ao seu perfil.	Confiabilidade
RNF06	Registro de ações (log)	O sistema deve registrar as ações realizadas pelos usuários, visando maior segurança e rastreabilidade.	Confiabilidade
RNF07	Integridade das informações	O sistema deve validar os campos durante operações de criação, alteração e exclusão de dados, assegurando o correto tipo de dado e o preenchimento de campos obrigatórios.	Confiabilidade
RNF08	Interface simples	O sistema deve possuir uma interface adequada ao perfil dos usuários, priorizando clareza e facilidade de uso.	Usabilidade

Código	Nome	Descrição	Categoria
RNF09	Organização das informações	As informações devem ser apresentadas de forma organizada, facilitando a leitura e compreensão pelo usuário.	Usabilidade
RNF10	Feedback ao usuário	O sistema deve informar o usuário sobre as ações realizadas, por meio de mensagens claras e objetivas.	Usabilidade

3.1.2 Fluxo do Usuário

O fluxo do usuário descreve as principais interações entre os diferentes perfis de usuários e as telas do sistema SISCAR, representando o caminho percorrido desde o acesso inicial até a execução das funcionalidades disponíveis. Essa modelagem tem como objetivo garantir uma navegação coerente, intuitiva e alinhada às necessidades operacionais da clínica.

No sistema SISCAR foram definidos três perfis de usuários distintos: professores, alunos e secretárias. Cada perfil possui permissões específicas, determinadas de acordo com suas responsabilidades, assegurando o controle de acesso às informações e funcionalidades do sistema.

Os professores possuem acesso completo às funcionalidades do sistema, incluindo o gerenciamento de usuários, pacientes, prontuários e consultas, além da supervisão das atividades realizadas pelos alunos. Os alunos têm acesso restrito às funcionalidades relacionadas ao atendimento clínico, podendo visualizar e registrar informações dos pacientes sob sua responsabilidade, bem como acessar o calendário de consultas. Já as secretárias possuem acesso limitado às funcionalidades de agendamento, sendo responsáveis pela organização e gerenciamento do calendário da clínica.

A definição clara do fluxo do usuário contribuiu para a organização das telas e para a definição dos caminhos de navegação do sistema, servindo como base para a prototipagem das interfaces e para a implementação do *frontend*.

3.1.3 Modelagem Lógica

A Modelagem Lógica (Figura 3) apresentado ilustra a estrutura do banco de dados do sistema SISCAR, evidenciando as principais entidades envolvidas no funcionamento da clínica, bem como seus atributos e relacionamentos. A modelagem foi construída com base nos requisitos levantados e na análise da ficha clínica utilizada na instituição, garantindo aderência entre o sistema e a realidade do atendimento odontológico acadêmico.

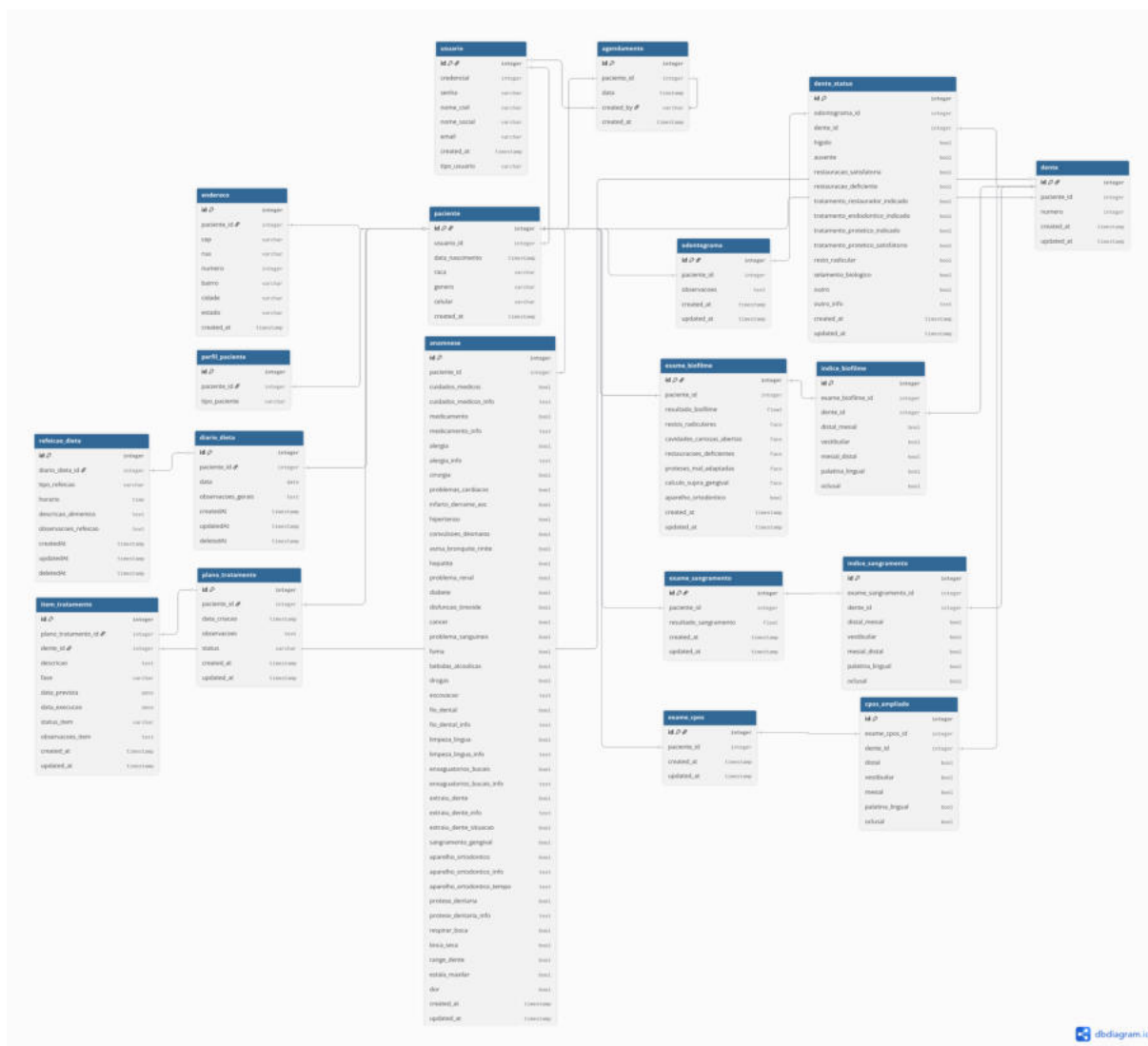


Figura 3 – Modelagem Lógica do sistema SISCAR

Fonte: Elaboração própria. Disponível em: <https://dbdiagram.io/d/Siscar-DB-67eaffde4f7afba184dbbdc9>. Acesso em: 27 fev. 2026.

A entidade central da modelagem é **Paciente**, que concentra as informações fundamentais do atendimento clínico. Essa entidade armazena dados pessoais do paciente e se relaciona diretamente com outras entidades que representam os diferentes registros clínicos realizados ao longo do tratamento, e também futuras implementações. Entre essas entidades destacam-se **Usuário**, **Anamnese**, **Dente**, **Odontograma**, **ExameBifilme**, **ExameSangramento**, **ExameCPOS**, **PlanoTratamento**, **Agendamento**, e **PerfilPaciente**.

A entidade **Usuário** representa os indivíduos que interagem com o sistema, contendo diferentes perfis, como *professores*, *alunos* e *secretárias*. Essa entidade armazena informações de autenticação e identificação, sendo fundamental para o controle de acesso

e rastreabilidade das ações realizadas no sistema. Além disso, essa modelagem foi pensada de forma a possibilitar evoluções futuras do sistema, nas quais o próprio paciente possa ter acesso à plataforma. Nesse contexto, todo paciente está associado a um usuário no sistema, mesmo que, no cenário atual, não possua acesso direto às funcionalidades da aplicação.

A **Anamnese** corresponde ao conjunto de informações clínicas e de saúde geral do paciente, coletadas antes do atendimento odontológico, com o objetivo de subsidiar a tomada de decisão clínica. Nesse registro são incluídos dados como histórico médico, uso de medicamentos, presença de alergias, doenças sistêmicas, hábitos de vida e condições que possam influenciar direta ou indiretamente o tratamento odontológico. No contexto do sistema SISCAR, a entidade **Anamnese** foi modelada de forma a permitir que um mesmo paciente possua uma ou múltiplas anamneses ao longo do tempo. Essa decisão se justifica pelo caráter dinâmico das informações clínicas, que podem se alterar conforme o paciente retorna à clínica em diferentes períodos. Condições de saúde que não estavam presentes em um primeiro atendimento podem ser diagnosticadas posteriormente, impactando diretamente a condução dos procedimentos odontológicos.

A entidade **Dente** representa cada um dos dentes pertencentes a um paciente. Em geral, os pacientes iniciam o acompanhamento clínico com a dentição permanente completa, composta por até 32 dentes, cada um identificado por uma numeração padronizada, que permite determinar o quadrante e a posição do dente na arcada dentária.

Com base nessa estrutura, os exames odontológicos foram modelados de forma hierárquica, distinguindo o exame como um conjunto de índices associados a cada dente. Essa abordagem possibilita maior flexibilidade, padronização dos registros e histórico clínico.

O **Odontograma** foi modelado como uma entidade que representa o registro visual e descritivo da condição dos dentes do paciente em um determinado momento. Cada odontograma é composto por vários registros de **DenteStatus**, sendo cada um associado a um dente específico. O status do dente contempla informações como presença ou ausência do dente, condições restauradoras, necessidade de tratamento restaurador, endodôntico ou protético, selamento biológico, entre outras observações clínicas relevantes.

O **ExameBiofilme** representa uma avaliação clínica realizada e está associado a um paciente. Esse exame é composto por diversos **ÍndiceBiofilme**, sendo cada índice relacionado a um dente específico. Os índices registram informações clínicas como presença de biofilme em diferentes faces do dente (vestibular, distal, mesial, palatina/lingual e oclusal).

De forma análoga, o **ExameSangramento** corresponde a uma avaliação do sangramento gengival do paciente. Esse exame é constituído por vários **ÍndiceSangramento**, cada um associado a um dente, registrando a presença ou ausência de sangramento em suas diferentes faces, com a diferença de que não há uma face oclusal nesse índice.

O **ExameCPOS** segue a mesma lógica. Ele representa o exame clínico geral do

paciente relacionado a dentes cariados, perdidos e obturados, sendo composto por diversos **ÍndicesCPOS**, cada um associado a um dente específico.

Tanto o odontograma quanto os exames odontológicos (biofilme, sangramento e CPOS) estão diretamente associados ao paciente e foram modelados de forma a permitir a existência de um ou mais registros ao longo do tempo, da mesma forma que a **Anamnese**. Essa decisão de modelagem considera que o paciente pode retornar à clínica em diferentes períodos, apresentando alterações em sua condição bucal.

A entidade **Agendamento** é responsável por representar os registros de marcação de atendimentos realizados no SISCAR. Essa entidade armazena informações essenciais para a organização dos atendimentos, como a data e o horário do procedimento, além das associações com o paciente envolvido e com o usuário responsável pela criação do agendamento.

A entidade **PerfilPaciente** foi criada para representar a classificação institucional dos pacientes atendidos na clínica. Esse perfil está relacionado a critérios administrativos e sociais definidos pela instituição, sendo atualmente dividido em *ação afirmativa (cotistas)*, *servidores da UFRJ* e *pacientes com doença falciforme*. Inicialmente, o perfil do paciente foi modelado como um atributo simples da entidade **Paciente**. No entanto, durante as reuniões de levantamento e refinamento de requisitos com os *stakeholders* da clínica, identificou-se que essa abordagem poderia se tornar limitada ao longo do tempo. Foi levantada a possibilidade de surgimento de novos perfis institucionais, assim como a eventual descontinuidade de perfis existentes, portanto foi criada uma entidade para modelar essa possibilidade.

O **DiárioDieta** foi modelado para representar o registro alimentar diário de um paciente, sendo composto por um conjunto de Refeições realizadas ao longo de um determinado dia. Cada diário está associado a um paciente e a uma data específica, permitindo o armazenamento de observações gerais relacionadas à alimentação naquele período.

A entidade **RefeiçãoDieta** representa cada refeição individual registrada dentro de um diário, contemplando informações como o tipo de refeição, horário, descrição dos alimentos consumidos e observações adicionais. Essa estrutura possibilita detalhar os hábitos alimentares do paciente de forma organizada e cronológica.

Apesar de o diário alimentar constar na ficha clínica utilizada como referência inicial para a modelagem, essa funcionalidade foi definida como uma implementação futura no sistema. Durante as reuniões com os *stakeholders* da CAR, foi identificado que, na prática atual, o registro alimentar não é utilizado de forma sistemática no atendimento clínico. No entanto, a modelagem foi mantida no banco de dados visando a evolução do sistema, permitindo que essa funcionalidade seja facilmente incorporada caso venha a ser adotada futuramente pela clínica.

O **PlanoTratamento** representa o conjunto de ações clínicas planejadas para um paciente após a realização da avaliação inicial, que envolve o levantamento do odontograma

e a execução dos exames clínicos. Essa entidade está diretamente associada ao paciente e armazena informações gerais do planejamento, como data de criação, status e observações relevantes. Cada plano de tratamento é composto por diversos **Itens Tratamento**, que representam as etapas ou procedimentos específicos a serem realizados. Cada item pode estar associado a um dente específico, quando aplicável, e contém informações como descrição do procedimento, fase do tratamento, datas previstas e executadas, status do item e observações clínicas.

O **Endereço** representa as informações de localização associadas a cada paciente atendido pela clínica, sendo responsável por armazenar dados essenciais como CEP, logradouro, número, complemento e bairro.

Inicialmente, os atributos referentes à cidade e ao estado eram representados como campos textuais dentro da própria entidade **Endereço**. No entanto, visando uma melhor organização dos dados, redução de redundâncias e maior integridade referencial, esses atributos foram posteriormente normalizados e transformados em entidades independentes. Dessa forma, foram criadas as tabelas **Cidade** e **Estado**, que passaram a ser relacionadas ao **Endereço** por meio de chaves estrangeiras.

Com a implementação das entidades e de seus respectivos relacionamentos, tornou-se possível visualizar de forma concreta a modelagem do banco de dados (Figura 4):

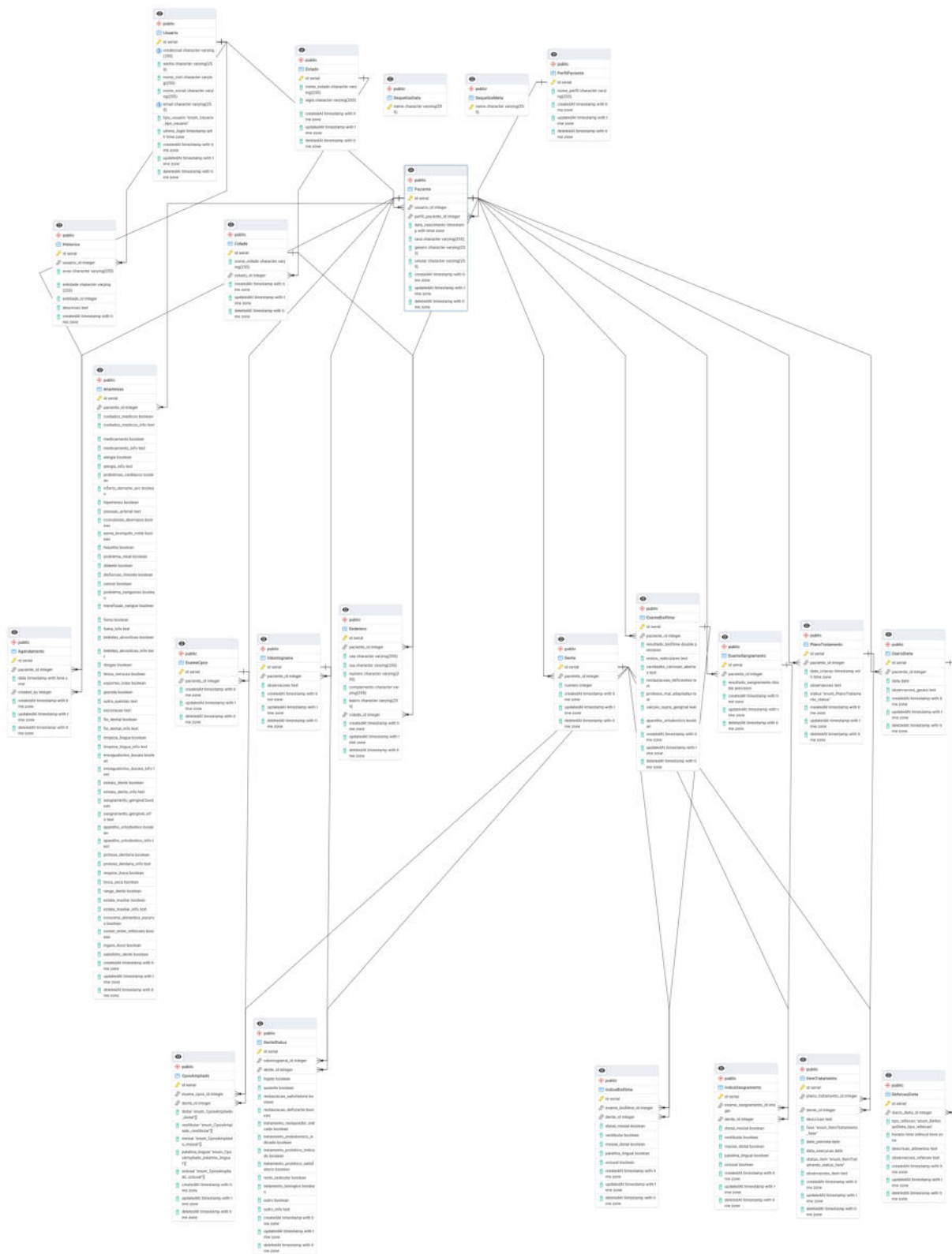


Figura 4 – Modelagem Física do sistema SISCAR

Fonte: Elaboração própria. Disponível em: <https://i.imgur.com/Wh2oeRp.png>. Acesso em: 27 fev. 2026.

3.1.4 Prototipagem de Interfaces com Figma

Após o levantamento e refinamento dos requisitos e a definição da modelagem do banco de dados, iniciou-se a etapa de prototipagem das interfaces do sistema SISCAR. Essa fase teve como objetivo principal representar visualmente as funcionalidades do sistema, permitindo a validação dos fluxos de navegação e da organização das informações antes da implementação.

A ferramenta Figma foi utilizada para a criação dos protótipos por possibilitar o desenvolvimento colaborativo de interfaces e a rápida iteração a partir de *feedbacks* dos usuários.

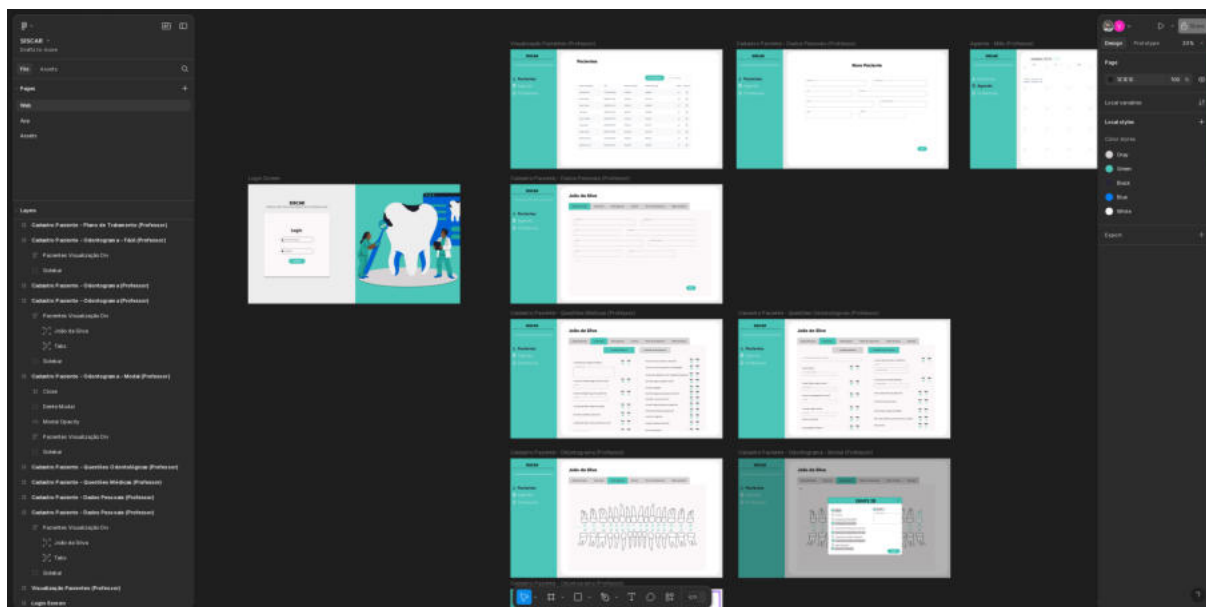


Figura 5 – Protótipo das interfaces do sistema SISCAR desenvolvido no Figma

O protótipo do sistema SISCAR pode ser acessado em: <https://www.figma.com/design/u4NEGMIKHkWZbn9dYZHiQ/SISCAR?node-id=0-1&t=1Mhg8ZXvkUMt2Jub-1>.

3.2 ARQUITETURA DA SOLUÇÃO PROPOSTA

Esta seção descreve a estrutura macroscópica do sistema SISCAR, detalhando como seus componentes se organizam, onde são executados e as tecnologias que sustentam a solução.

3.2.1 Componentes Principais e Inter-relações

A solução foi estruturada seguindo o modelo cliente-servidor, composta por três camadas principais que interagem de forma coordenada:

- **Frontend (Interface do Usuário):** Responsável pela interação direta com professores, alunos e secretárias. Ele consome os recursos da API e gere o estado local da aplicação.
- **Backend (API RESTful):** Concentra a lógica de negócio, as regras de validação e o controle de acesso (autenticação/autorização). Ele atua como o mediador entre as requisições do usuário e o armazenamento de dados.
- **Banco de Dados (Persistência):** Camada responsável pelo armazenamento íntegro e relacional de todas as informações clínicas e administrativas.

As inter-relações ocorrem via protocolo HTTP, onde o Frontend envia requisições formatadas em JSON para o Backend. Este, por sua vez, processa a solicitação, interage com o banco de dados via ORM e devolve uma resposta estruturada, garantindo que a interface permaneça sempre sincronizada com o estado do servidor.

3.2.2 Arquitetura de Sistema e Instanciamento

A distribuição física e lógica dos componentes foi projetada para garantir portabilidade e isolamento:

- **Componentes de Lógica de Apresentação:** São instanciados no *client-side*. No caso do acesso *web*, o navegador do usuário carrega e executa o código compilado; no caso de dispositivos móveis, a instância ocorre localmente no sistema operacional (Android ou iOS).
- **Componentes de Lógica de Negócio e Persistência:** São instanciados no *server-side*. Para garantir a paridade entre ambientes, esses componentes são executados dentro de contêineres Docker, isolando o *runtime* do Node.js e o serviço do PostgreSQL do sistema operacional hospedeiro.

3.2.3 Tecnologias de Implementação

Para a materialização da arquitetura proposta, foram selecionadas tecnologias de código aberto com ampla adoção na comunidade técnica, conforme resumido no Quadro 4:

Quadro 4 – Tecnologias utilizadas na implementação do SISCAR

Camada	Tecnologia	Função Principal
Frontend	Flutter (Dart)	Desenvolvimento multiplataforma e UI responsiva.
Backend	Node.js (Express)	Criação da API RESTful e lógica de servidor.
Banco de Dados	PostgreSQL	Armazenamento relacional e integridade de dados.
Infraestrutura	Docker	Containerização e padronização de ambientes.
Testes	Jest	Automação de testes unitários e de integração.

3.3 FRONTEND

3.3.1 O porquê do uso do Flutter

Após a definição dos requisitos e o início da prototipagem, optou-se por utilizar o *Flutter* como principal tecnologia para o desenvolvimento do sistema SISCAR. Essa decisão foi baseada em critérios técnicos e nas necessidades específicas do projeto, especialmente no que se refere à multiplataforma e à consistência da interface do usuário.

O *Flutter* (Versão: 3.38.5) é um *framework* de desenvolvimento de aplicações multiplataforma de código aberto, mantido pelo **Google**, que permite que uma única base de código seja utilizada para gerar aplicações em dispositivos móveis (**Android e iOS**), *web* e *desktops* (**Windows, macOS e Linux**), o que reduz significativamente o tempo de desenvolvimento e a complexidade de manutenção em comparação com abordagens nativas separadas para cada plataforma. (SONAWANE, 2025)

Adicionalmente, o *Flutter* oferece um alto desempenho, pois compila diretamente para código de máquina utilizando a linguagem *Dart*, sem depender de pontes entre o código da interface e o sistema operacional subjacente. Estudos de benchmark, como os realizados por (GOOGLE,), demonstram que o Flutter mantém uma taxa de atualização de 60 FPS (quadros por segundo) de forma mais consistente que competidores como o React Native, resultando em uma experiência de usuário (*UX*) superior e livre de gargalos de processamento.

Outro aspecto importante é a facilidade de desenvolvimento e prototipagem proporcionada pela funcionalidade de **Hot Reload**, que permite visualizar alterações no aplicativo em tempo real sem reiniciar o processo de execução. Essa característica acelera a iteração entre design e implementação, favorecendo ciclos rápidos de refinamento e testes durante toda a fase de desenvolvimento. (SONAWANE, 2025)

Por fim, o forte suporte da comunidade de desenvolvedores e o contínuo investimento do **Google** garantem que o *Flutter* permaneça atualizado, com um ecossistema crescente de pacotes, ferramentas e documentação, o que contribui para a sustentabilidade técnica do projeto a longo prazo. (EWEBWORLD, 2025)

Dessa forma, a escolha pelo *Flutter* não apenas atende às exigências de multiplataforma e responsividade do SISCAR, como também proporciona vantagens de desempenho, produtividade no desenvolvimento e manutenção facilitada, alinhando-se às melhores práticas atuais em desenvolvimento de aplicações modernas.

3.3.2 Arquitetura e Estrutura de Pastas

A adoção de uma arquitetura de *frontend* bem definida foi motivada não apenas por critérios técnicos, mas também pela preocupação com a continuidade, manutenção e evolução do projeto ao longo do tempo. Considerando que o sistema não será desenvolvido

indefinidamente pelos autores, tornou-se essencial estruturar o código de maneira que desenvolvedores futuros possam compreender a aplicação facilmente. Com esse objetivo foi iniciado um processo evolutivo de decisões, começando pela estruturação macro do sistema e refinando-se na organização das interfaces. Inicialmente, estabeleceu-se uma separação baseada nos princípios da *Clean Architecture* (MARTIN, 2017), organizando o código em três camadas fundamentais para garantir o desacoplamento e a robustez do sistema:

- **Core:** Concentra a infraestrutura transversal e os componentes reutilizáveis, como configurações globais, serviços compartilhados (autenticação, rotas), tema visual e widgets comuns.
- **Data:** Responsável pelo acesso a dados e integrações externas, englobando modelos (*models*), serviços de comunicação com a *API* e repositórios.
- **Presentation:** Destinada exclusivamente à interface gráfica e ao gerenciamento do estado das telas.

Uma vez consolidada essa estrutura global, o foco voltou-se para a organização interna da camada de *Presentation*. A medida que o projeto foi tomando forma a quantidade de arquivos nesta pasta foi crescendo exponencialmente, foi neste contexto que surgiu a necessidade de uma abordagem que facilitasse não apenas o desenvolvimento imediato, mas principalmente a manutenção futura. Adotou-se, então, o padrão *MVVM* (*Model-View-ViewModel*) (GOSSMAN, 2005) com organização *Feature-based* (baseada em funcionalidades).

A organização feature-base contribui diretamente para esse objetivo, uma vez que cada módulo do sistema é relativamente autocontido. Funcionalidades pertinentes a agenda, paciente ou professor podem ser analisadas de forma independente, sem a necessidade de navegar por múltiplas pastas técnicas distribuídas pelo projeto. Essa característica é especialmente relevante em cenários de manutenção corretiva ou evolutiva, nos quais é necessário atuar rapidamente em partes específicas do sistema, minimizando o risco de impactos colaterais em outras funcionalidades.

O padrão *MVVM*, por sua vez, reforça ainda mais essa facilidade de continuidade, com uma separação clara entre interface gráfica, estado e regras de apresentação. As *View* são responsáveis exclusivamente pela construção da interface e pelo tratamento de eventos de interação, enquanto os *ViewModels* concentram o estado da tela e a lógica necessária para o funcionamento das funcionalidades. Essa divisão torna o fluxo da aplicação mais previsível, melhora a legibilidade do código e evita a mistura de responsabilidades. Para quem assume o projeto no futuro, essa organização reduz significativamente o tempo necessário para compreender o comportamento das telas e identificar com precisão os pontos onde correções ou melhorias devem ser realizadas.

Em suma, a arquitetura final é o resultado de duas decisões complementares: a *Clean Architecture* garantiu a solidez estrutural e a separação de responsabilidades entre infraestrutura e dados, enquanto o *MVVM Feature-based* na camada de apresentação assegurou que a interface do usuário permanecesse modular, legível e preparada para evoluir nas mãos de futuros desenvolvedores.

3.3.3 Responsividade

Desde o início do desenvolvimento, foi estabelecido como requisito que a aplicação fosse acessível tanto em dispositivos móveis quanto em ambiente *web*, o que tornou a responsividade um aspecto central da arquitetura de interface. Para atingir esse objetivo, foram combinadas múltiplas estratégias complementares. Inicialmente, o sistema realiza a detecção do tipo de dispositivo e do tamanho de tela, classificando o ambiente em três categorias principais a partir de *breakpoints* definidos no projeto:

- **Mobile:** Largura menor que 600px
- **Tablet:** Largura entre 600px e 1024px
- **Desktop:** Largura maior ou igual a 1024px

Esta classificação é baseada no menor lado da tela, garantindo comportamento consistente independente da orientação (especialmente útil em dispositivos móveis onde a rotação altera significativamente o espaço disponível). Com base nessa classificação o componente *Base*, que atua como *wrapper* para todas as telas do sistema (exceto *Login*, *Splashscreen* e telas de erro), decide em tempo de execução, qual *layout* deve ser renderizado. Os componentes de navegação também são adaptados a cada contexto:

- **MobileSidebar:** *Drawer* deslizante para telas pequenas
- **TabletSidebar:** Barra lateral compacta para dispositivos intermediários
- **DesktopSidebar:** Navegação permanente para telas grandes

Além da escolha de layouts específicos, o projeto utiliza o pacote *ScreenUtil* para realizar o escalonamento proporcional de tamanhos, assegurando que espaçamentos, dimensões e tipografia mantenham proporções adequadas mesmo em resoluções muito diferentes. Esta abordagem evita a dependência de valores fixos e contribui para uma interface mais consistente em telas pequenas e grandes.

Essa preocupação se estende à componentes de maior complexidade funcional e visual. Os exemplos mais claro dessas adaptações contextuais estão no formulário para cadastro de um novo paciente. Desde o indicador de etapas do formulário ao Odontograma onde:

- **Desktop:** Utiliza *SVGs* interativos e *painters* detalhados, priorizando precisão visual e interação com mouse
- **Mobile:** Adaptado para grids retangulares e *painters* simplificados, otimizados para toque e restrições de espaço

Dessa forma, a responsividade no projeto não é apenas uma reação ao tamanho da janela, mas uma decisão ativa de design. Ao centralizar essas regras no *Base*, o sistema garante que novos desenvolvedores não precisem reimplementar lógica de adaptação a cada nova tela, bastando fornecer as *views* correspondentes para que a arquitetura cuide da coerência visual e da navegação. Por fim, para que essa adaptação visual e comportamental ocorra de maneira uniforme em toda a aplicação, as decisões de responsividade são sustentadas por um *Design System* baseado em *tokens*. Cores, tipografia, espaçamentos, raios de borda e variações de tema são definidos de forma centralizada e reutilizados tanto nos layouts *mobile* quanto *desktop*, assegurando que a interface permaneça coerente mesmo quando sua estrutura se adapta ao contexto. Essa integração entre responsividade e *design system* permite que a aplicação evolua visualmente sem comprometer a consistência, servindo como base para a organização temática e visual apresentada na seção seguinte.

3.3.4 Temas e Tokens

Para sustentar a responsividade descrita anteriormente e garantir que a interface mantenha sua consistência visual independentemente da escala ou do dispositivo, o SISCAR adota uma arquitetura de *Design System* baseada em *Tokens* (FILHO, 2020). Essa abordagem desacopla as decisões de estilo (cores, tipografia, formas) da implementação dos componentes, estabelecendo uma “fonte de verdade” única para a identidade visual da aplicação. Evitando, assim, “números mágicos” espalhados ou códigos hexadecimais pelas telas. O sistema de cores é construído com base no Material Design 3, utilizando cores primárias, secundárias, neutras e estados semânticos (como sucesso, erro, aviso e informação), baseadas no Figma montado anteriormente e contemplando variações para os modos claro e escuro. Foi padronizado a tipografia utilizando a fonte Poppins através do pacote `google_fonts` em toda a aplicação e foi definido uma escala tipográfica numérica para títulos, corpo de texto e labels, assegurando hierarquia visual clara. Também foi estabelecido escalas consistentes de espaçamento e arredondamento de bordas, aplicáveis a botões, cards, formulários e diálogos. Empregam-se esses *tokens* na montagem do tema, tanto para o modo claro quanto para o modo escuro. A partir desse tema, componentes visuais passam a herdar estilos coerentes por padrão, reduzindo a necessidade de configurações locais e evitando divergências visuais entre *features*. Essa arquitetura garante que a evolução visual do projeto seja centralizada. Uma alteração no raio da borda ou

uma mudança de tom propaga-se instantaneamente por toda a aplicação, reforçando a manutenibilidade e a escalabilidade do *frontend*.

3.3.5 Tratamento de Erros

O tratamento de erros no SISCAR foi projetado para transformar falhas técnicas em comportamentos previsíveis e semânticos, com o objetivo de facilitar manutenção e oferecer uma experiência de usuário consistente diante de falhas. Em vez de tratar erros pontualmente em cada tela ou chamada de rede, a aplicação adota uma estratégia em camadas, na qual cada nível possui responsabilidades bem definidas.

3.3.5.1 Tipagem e Semântica de Exceções

O sistema é estruturado em torno da classe base `AppException`, que fornece uma estrutura padronizada para todos os erros da aplicação. A partir dessa base, deriva-se uma hierarquia granular que permite à aplicação reagir distintamente a diferentes cenários:

- **Infraestrutura:** `NetworkException`, `RequestTimeoutException`.
- **Segurança:** `UnauthorizedException`, `TokenExpiredException`.
- **Domínio/Dados:** `InvalidModelException`, `ParsingException`.

Essa taxonomia elimina a ambiguidade. O sistema não recebe apenas um “erro”; ele sabe diferenciar um erro de *parsing JSON* (`JsonParsingException`) de uma regra de negócio violada, permitindo tratamentos específicos (ex: *logout* forçado em caso de `AuthException` vs. *retry* em caso de `NetworkException`).

3.3.5.2 Normalização na Camada de Serviço

Na camada de serviços, o tratamento de erros ocorre no ponto mais próximo da comunicação com a *API*. O `ApiClient` atua como a primeira barreira de defesa interceptando todas as respostas *HTTP* e normalizando comportamentos de sucesso e falha. Respostas com códigos de erro (4xx e 5xx) são convertidas em exceções específicas de domínio, como `UnauthorizedException`, `NotFoundException`, `RequestTimeoutException` e `MalformedResponseException`. Isso garante que as camadas superiores (Repositórios e `ViewModels`) nunca lidem com objetos `http.Response` brutos ou *JSONs* malformados. Se o *backend* retorna um 403, o `ApiClient` lança imediatamente uma `UnauthorizedException`, abstraindo a complexidade do protocolo *HTTP*.

3.3.5.3 Camada de Repositório: Contextualização

Na camada de repositório, os erros provenientes dos serviços já tipados são recebidos e, em vez de serem simplesmente repassados, são contextualizados sob a perspectiva do

uso específico dos dados. Exceções que já possuem significado claro, como uma *UnauthorizedException*, são propagadas sem alteração, preservando sua intenção original. Já falhas inesperadas, como erros de conversão durante o mapeamento de *JSON* para modelos de domínio, são encapsuladas em novas exceções que refletem o contexto da operação – por exemplo, um *FormatException* durante o *parsing* de uma lista de agendamentos torna-se uma *ParsingException* com detalhes sobre o campo “agendamentos”. Este processo garante que a camada de apresentação permaneça isolada de detalhes de implementação de baixo nível, como estruturas *JSON* específicas ou particularidades da comunicação *HTTP*. Essa abordagem permite, inclusive, a introdução futura de estratégias adicionais, como *cache* local ou tentativas de recuperação (*retry*), sem impacto nos *ViewModels*.

3.3.5.4 Camada de ViewModel: Gestão de Estado

Na camada de *ViewModel*, os erros passam a ser tratados como parte do estado da tela. As operações assíncronas são protegidas por blocos *try/catch*, e exceções capturadas são armazenadas em propriedades observáveis, permitindo que a *UI* reaja de forma declarativa. O *ViewModel* não apenas armazena o erro, mas decide qual será seu impacto na experiência do usuário: se deve bloquear completamente a interação, exibir uma mensagem transitória que permite continuidade, ou exigir uma ação corretiva, como um novo *login*, mantendo a lógica de decisão fora das *Views*. Essa separação evita que telas implementem regras próprias de tratamento e métodos como *clearError()* permitem resetar o estado de erro após o usuário tomar ciência do problema, completando o ciclo.

3.3.5.5 Camada de View: Apresentação ao Usuário

A camada de *View* é responsável apenas pela apresentação do erro ao usuário, sem interpretar sua origem ou natureza técnica. Para isso, o projeto utiliza componentes reutilizáveis, que exibem mensagens claras, títulos padronizados e, quando necessário, códigos de erro para rastreabilidade. O *mixim ErrorHandlerMixin*, utilizado nas *States* das telas, centraliza a lógica de exibição: ele observa a propriedade de erro do *ViewModel* e, quando detecta um valor, invoca o componente de apresentação apropriado. Para erros contextuais que requerem reconhecimento imediato – como falha ao salvar um formulário – é utilizado o *ErrorPopup*, um diálogo modal personalizado que exibe título, mensagem e código do erro, impedindo interação com o restante da interface até ser dispensado. Para falhas que impedem completamente a renderização de uma rota – como páginas não encontradas (404), acesso negado (403) ou erros internos do servidor (500) – são empregadas telas completas construídas com o componente *ErrorScaffold*, que oferece uma experiência robusta com código de erro, título, mensagem explicativa, imagem ilustrativa e uma ação primária clara, como voltar à tela anterior. Além disso, a navegação é protegida pela classe *RouteUtils*. Todas as transições de tela são encapsuladas para capturar *RouteBu-*

ildingException, garantindo que URLs malformadas ou parâmetros inválidos não causem o “crash” da aplicação, mas sim um *feedback* visual controlado.

Esta arquitetura em camadas não apenas torna o código mais legível e testável, como também estabelece uma base sólida para evoluções futuras, como a integração com sistemas de monitoramento (Sentry, Crashlytics) ou a implementação de padrões avançados de resiliência, sem exigir refatorações invasivas nas telas ou na lógica de negócio da aplicação.

3.3.6 Persistência e sessão

A estratégia de persistência do SISCAR é deliberadamente restrita, com foco exclusivo na manutenção da sessão de autenticação entre reinicializações da aplicação. Essa decisão evita o acúmulo de estado local complexo e reduz riscos de inconsistência, ao mesmo tempo em que garante uma experiência de uso contínua para o usuário autenticado.

3.3.6.1 Infraestrutura de Armazenamento

Essa responsabilidade está centralizada em `core/storage/auth_storage.dart`, por meio da classe `AuthStorage`, que encapsula o acesso ao mecanismo de armazenamento seguro da plataforma. São persistidos apenas os dados estritamente necessários ao ciclo de vida da sessão: o usuário autenticado, o *token* de acesso e, opcionalmente, a credencial lembrada para facilitar o *login*. Ao isolar completamente essa lógica, o restante da aplicação interage com autenticação em um nível conceitual, sem depender de detalhes de persistência ou tecnologia subjacente.

3.3.6.2 Ciclo de Vida da Sessão

Durante a inicialização da aplicação, ainda no `main()`, o `AuthService` consulta o `AuthStorage` para verificar a existência de uma sessão previamente válida. Caso os dados persistidos estejam íntegros, o usuário é restaurado automaticamente e a aplicação já inicia em estado autenticado, evitando telas intermediárias e garantindo que a navegação e as permissões sejam configuradas com base em um estado consistente desde o primeiro frame. Essa restauração de sessão não ocorre de forma isolada: ela está diretamente ligada ao gerenciamento de escopos da injeção de dependências. Quando uma sessão válida é detectada, o `AuthService` aciona explicitamente a abertura do escopo de sessão no `container` de dependências. Esse escopo passa a concentrar repositórios e serviços que dependem do usuário autenticado, como os módulos de agenda, pacientes, professores e secretarias. De forma complementar, no `logout` — ou em casos de falha na leitura do `storage` — o escopo de sessão é encerrado, garantindo que objetos sensíveis e estados associados ao usuário sejam corretamente descartados.

3.3.6.3 Segurança e Robustez

Do ponto de vista de segurança, o armazenamento é realizado com *FlutterSecureStorage*, que utiliza mecanismos nativos de criptografia nas plataformas móveis. Além disso, o *token* é tratado como um dado efêmero: falhas de leitura, dados corrompidos ou inconsistências levam à limpeza preventiva do *storage* e à exigência de nova autenticação. Essa estratégia evita que a aplicação permaneça em estados inválidos ou parcialmente autenticados. A separação entre dados persistidos (sessão) e dados voláteis (estado de telas, formulários e entidades de domínio) reforça o papel do *storage* como elemento de infraestrutura, e não como *cache* de domínio. Pacientes, agendamentos e exames são sempre carregados via *API* e mantidos apenas em memória, o que simplifica o controle de estado e reduz a complexidade do *frontend*. Dessa forma, persistência e injeção de dependências operam de maneira integrada: o *storage* define se existe uma sessão válida, enquanto o *container* de dependências define quais serviços e repositórios devem existir em função dessa sessão. Essa combinação garante isolamento entre usuários, previsibilidade no ciclo de vida dos objetos e uma base sólida para manutenção e evolução do sistema.

3.3.7 Injeção de Dependências e Gerenciamento de Escopos

A distinção entre estado persistido e estado volátil apresentada na seção anterior exige um mecanismo capaz de controlar com precisão o ciclo de vida dos objetos da aplicação. No SISCAR, essa responsabilidade é assumida pela injeção de dependências, que atua como o elo entre autenticação, sessão e organização das camadas, garantindo que serviços e repositórios existam apenas enquanto forem válidos no contexto do usuário atual. Para isso, a aplicação adota o *GetIt* como *container* de injeção de dependências, centralizando o registro e a resolução de serviços, repositórios e *ViewModels*. Essa escolha permite desacoplar completamente as classes consumidoras de suas implementações concretas, facilitando testes, manutenção e evolução do sistema. O *container* é estruturado em escopos explícitos, refletindo diretamente o ciclo de vida da sessão descrito anteriormente. No escopo global, são registrados componentes independentes do usuário autenticado, como serviços de inicialização, autenticação e navegação. Já o escopo de sessão é aberto apenas após a restauração ou criação de uma sessão válida e concentra dependências que estão semanticamente ligadas ao usuário logado, como repositórios de pacientes, agenda e usuários do sistema. Esse gerenciamento explícito de escopos garante que, ao encerrar a sessão (logout ou falha de autenticação), todos os objetos associados sejam descartados de forma controlada, evitando vazamento de estado entre usuários e reduzindo riscos de inconsistência. Além disso, o uso de registros do tipo *lazySingleton* assegura que dependências sejam instanciadas apenas quando necessárias, contribuindo para desempenho e clareza arquitetural. A partir dessa base, *ViewModels* e camadas superiores passam a depender apenas de contratos bem definidos, sem conhecimento direto sobre criação, ciclo de vida

ou dependências internas dos serviços utilizados. Essa organização reforça o isolamento entre camadas e sustenta a arquitetura *MVVM* adotada, permitindo que funcionalidades evoluam de forma previsível e segura ao longo do tempo.

3.4 BACKEND

3.4.1 O porquê do uso do Node.js com Express

O *backend* do sistema SISCAR foi desenvolvido utilizando o framework *Express*, executado sobre *Node.js*. A escolha dessa tecnologia se deu pela sua adequação ao desenvolvimento de APIs *RESTful*, que permitem a comunicação eficiente entre o *frontend*, desenvolvido em Flutter, e o banco de dados do sistema.

O *Express* oferece uma arquitetura leve e flexível, facilitando a separação de responsabilidades entre as camadas da aplicação e promovendo o desacoplamento entre cliente e servidor (LAXAAR, 2024). Essa abordagem contribui para a manutenção, escalabilidade e evolução do sistema, aspectos relevantes em um ambiente acadêmico que envolve múltiplos perfis de usuários, como alunos, professores, secretaria e pacientes.

Além disso, o *Node.js* adota um modelo de execução assíncrono e orientado a eventos, o que permite lidar de forma eficiente com múltiplas requisições simultâneas. Essa característica é especialmente adequada para sistemas *web* e móveis, como o SISCAR, que demandam respostas rápidas e acesso concorrente aos dados clínicos.

Outro fator determinante na escolha do *Express* foi seu amplo ecossistema de bibliotecas e a facilidade de integração com ferramentas de autenticação, validação de dados e acesso a bancos relacionais. Essas características favorecem a produtividade no desenvolvimento e se alinham à metodologia ágil adotada no projeto, permitindo a evolução incremental das funcionalidades ao longo das sprints.

3.4.2 Arquitetura e Estrutura de Pastas

A arquitetura do *backend* do sistema SISCAR foi organizada com base no padrão arquitetural *Model-View-Controller (MVC)*, amplamente discutido na literatura como uma abordagem que promove a separação de responsabilidades, modularidade e facilidade de manutenção do *software*. Esse padrão já foi apresentado anteriormente no referencial teórico, sendo aqui aplicado de forma adaptada ao contexto de uma *API*.

No SISCAR, o *backend* desenvolvido com *Node.js* e *Express* atua exclusivamente como uma *API RESTful*, responsável por fornecer serviços e dados ao *frontend* desenvolvido em Flutter. Dessa forma, não há geração de interfaces visuais no lado do servidor, o que implica na ausência da camada *View* tradicionalmente presente no padrão *MVC*. Assim, o padrão é utilizado de forma parcial, concentrando-se nas camadas *Model* e *Controller*.

A camada **Model** é responsável pela representação dos dados e pela comunicação direta com o banco de dados PostgreSQL. Nessa camada estão implementadas as operações

de acesso, persistência e recuperação das informações clínicas, respeitando a modelagem conceitual definida para o sistema. Essa separação permite que regras relacionadas aos dados permaneçam isoladas da lógica de controle das requisições.

A camada **Controller** atua como intermediária entre as requisições *HTTP* recebidas pela *API* e os modelos de dados. Os controladores são responsáveis por processar as requisições, aplicar validações, invocar os métodos apropriados da camada de modelos e retornar as respostas no formato *JSON* ao cliente. Essa organização contribui para um fluxo claro de responsabilidades e facilita a manutenção e evolução do código.

Essa abordagem arquitetural favorece a escalabilidade do sistema, a reutilização de código e a clareza estrutural do projeto, além de facilitar o trabalho colaborativo e a aplicação de práticas ágeis durante o desenvolvimento. Dessa forma, a adoção do *MVC*, mesmo em sua forma adaptada, mostrou-se adequada às necessidades do sistema e ao contexto acadêmico em que o SISCAR está inserido.

3.4.3 Segurança e Controle de Acesso

A segurança do sistema SISCAR foi projetada considerando a proteção dos dados clínicos, o controle de acesso aos recursos e a autenticação adequada dos diferentes perfis de usuários envolvidos na clínica. Para atender a esses requisitos, foi adotado o mecanismo de autenticação baseado em *JSON Web Token (JWT)*, amplamente utilizado em arquiteturas orientadas a serviços e APIs *RESTful*.

O *JWT* é um padrão aberto que permite a troca segura de informações entre cliente e servidor por meio de *tokens* assinados digitalmente. No SISCAR, após a autenticação bem-sucedida do usuário, é gerado um *token* contendo informações essenciais de identificação, o qual é enviado ao cliente e armazenado para uso nas requisições subsequentes. Esse *token* é então incluído no cabeçalho das requisições *HTTP*, permitindo que o *backend* valide a identidade do usuário sem a necessidade de manter sessões no servidor, caracterizando uma abordagem *stateless*.

A validação do *token* ocorre a cada requisição protegida, verificando sua assinatura, integridade e tempo de expiração. Essa estratégia reduz riscos associados ao acesso não autorizado e garante que apenas usuários autenticados possam acessar os recursos do sistema. O uso de expiração de *tokens* também contribui para a mitigação de riscos em caso de comprometimento de credenciais.

Além da autenticação, o SISCAR implementa um controle de acesso baseado em papéis (*Role-Based Access Control*). Nesse modelo, as permissões de acesso são definidas de acordo com o perfil do usuário, como professores, alunos, secretaria e pacientes. Cada perfil possui níveis distintos de acesso às funcionalidades e aos dados do sistema, assegurando que ações sensíveis sejam realizadas apenas por usuários autorizados.

Essa separação de permissões é fundamental no contexto de uma clínica, onde diferentes atores participam do processo clínico e pedagógico. Em conjunto, o uso de *JWT*

para autenticação e o controle de acesso por perfis contribuem para a confidencialidade, integridade e rastreabilidade das informações armazenadas no sistema. Essa abordagem fortalece a segurança do SISCAR e está alinhada às boas práticas de desenvolvimento de sistemas *web* modernos, especialmente em ambientes que lidam com dados sensíveis e múltiplos níveis de acesso.

3.4.4 Integração com o banco de dados PostgreSQL

A integração entre o *backend* do sistema SISCAR e o banco de dados PostgreSQL foi realizada por meio do uso de um *ORM* (*Object-Relational Mapping*), especificamente o *Sequelize*. A adoção de um *ORM* tem como objetivo abstrair o acesso direto ao banco de dados, permitindo que as operações fundamentais de *CRUD* (*Create, Read, Update e Delete*) sejam realizadas por meio de objetos e métodos, reduzindo a dependência de consultas *SQL* explícitas e facilitando a manutenção do sistema.

O *Sequelize* é um *ORM* amplamente utilizado no ecossistema *Node.js* e oferece suporte nativo ao PostgreSQL, além de recursos que contribuem para a organização do código e a integridade dos dados. Sua estrutura permite a definição clara de modelos que representam as entidades do domínio do sistema, alinhando-se à modelagem conceitual previamente estabelecida para o SISCAR.

Um dos fatores determinantes para a escolha do *Sequelize* foi o suporte à funcionalidade conhecida como *paranoid*, que implementa o conceito de exclusão lógica de registros. Nesse modelo, os dados não são removidos fisicamente do banco de dados quando excluídos, sendo apenas marcados como inativos por meio de um atributo de controle temporal. Essa abordagem é particularmente relevante no contexto de uma clínica, onde a preservação do histórico clínico é fundamental para fins acadêmicos, legais e de auditoria, evitando a perda definitiva de informações sensíveis, mesmo em operações de exclusão realizadas no âmbito do *CRUD*.

Além disso, o *Sequelize* foi utilizado em conjunto com o mecanismo de *migrations*, responsável pelo versionamento e controle da estrutura do banco de dados PostgreSQL. As *migrations* permitem a criação, alteração e remoção de tabelas e atributos de forma controlada e reproduzível, garantindo que diferentes ambientes de execução mantenham a mesma estrutura de dados ao longo do ciclo de desenvolvimento.

Também foram utilizados *seeds* para a inicialização do banco de dados com informações essenciais ao funcionamento do sistema. Esse processo possibilita que o ambiente seja iniciado já com dados previamente definidos, como um usuário do tipo professor, cujas credenciais de acesso são configuradas por meio de variáveis de ambiente (*.env*), além do cadastro inicial dos perfis de pacientes, estados e cidades. Essa estratégia contribui para a padronização do ambiente de desenvolvimento, facilita testes e demonstrações do sistema e reduz erros decorrentes da ausência de dados fundamentais para a operação inicial da aplicação.

Outro recurso relevante fornecido pelo *Sequelize* é o uso de *hooks*, que permitem a execução automática de funções em momentos específicos do ciclo de vida dos dados, como antes ou após operações de criação, atualização ou exclusão. Esses mecanismos são utilizados para reforçar regras de negócio e segurança, como validações adicionais, tratamento de dados sensíveis e padronização de comportamentos, contribuindo para a consistência das operações *CRUD* e para a redução da duplicação de código.

No contexto do sistema SISCAR, os *hooks* foram aplicados, por exemplo, na entidade **Paciente**, cuja ficha clínica não pode ser removida de forma direta, mas apenas editada. No entanto, quando ocorre a exclusão de um paciente é necessário garantir que todas as informações diretamente associadas a esse paciente sejam tratadas de forma consistente. Para atender a essa necessidade, foi implementado um *hook* do tipo *afterDestroy*, responsável por propagar a exclusão para as entidades relacionadas, como usuário vinculado, agendamentos, anamneses, diário alimentar, dentes, odontogramas, planos de tratamento e exames clínicos. Esse procedimento assegura a integridade referencial do banco de dados e evita a permanência de registros órfãos, mantendo a coerência das informações armazenadas.

3.5 DOCKER E CONTAINERIZAÇÃO

A utilização de *Docker* (INC., 2026) no desenvolvimento do sistema SISCAR teve como principal objetivo padronizar o ambiente de execução da aplicação, garantindo maior consistência entre os ambientes de desenvolvimento, testes e produção. A containerização permite que toda a infraestrutura necessária para a execução do sistema, incluindo *backend*, banco de dados e dependências seja encapsulada em contêineres isolados, reduzindo problemas relacionados a incompatibilidades de versões ou configurações específicas de cada máquina.

Por meio do uso de imagens e contêineres *Docker*, tornou-se possível inicializar o ambiente do sistema de forma rápida e reproduzível, facilitando tanto o desenvolvimento colaborativo quanto a implantação da aplicação. Além disso, a containerização favorece a manutenção e a escalabilidade do sistema, permitindo que novos serviços ou ajustes na infraestrutura sejam realizados com menor impacto no funcionamento geral da aplicação.

Outro aspecto fundamental dessa abordagem foi o uso de variáveis de ambiente, responsáveis por centralizar configurações sensíveis e parâmetros específicos de execução, evitando que tais informações fiquem acopladas ao código-fonte. No sistema SISCAR, as variáveis de ambiente foram utilizadas para definir o modo de execução da aplicação, portas de comunicação, credenciais administrativas iniciais, configurações de autenticação e parâmetros de conexão com o banco de dados PostgreSQL.

Entre as principais variáveis configuradas estão aquelas relacionadas ao ambiente de execução do *backend*, como o modo de operação da aplicação e a porta utilizada pelo

servidor, além das credenciais iniciais do usuário administrador (professor), empregadas para a inicialização controlada do sistema. Também foram definidas variáveis para a configuração do mecanismo de autenticação baseado em *JSON Web Token (JWT)* (IETF, 2026), incluindo a chave secreta e o tempo de expiração dos *tokens*, assegurando o controle de sessões e a proteção das rotas do sistema.

3.6 TESTES

A etapa de testes do sistema SISCAR foi conduzida com o objetivo de garantir o correto funcionamento das funcionalidades implementadas, bem como a confiabilidade das regras de negócio definidas ao longo do desenvolvimento. Para a realização dos testes automatizados, foi utilizada a ferramenta *Jest*, amplamente adotada no ecossistema *JavaScript*, que possibilita a validação de comportamentos esperados por meio de testes unitários e de integração.

A Figura 6 apresenta os resultados da execução dos testes automatizados e a organização da suíte de testes do projeto.

```

PASS tests/plano_tratamento-item.test.js
PASS tests/diario_dieta-refeicao.test.js
PASS tests/auth.test.js
PASS tests/paciente.test.js
PASS tests/professor.test.js
PASS tests/secretaria.test.js
PASS tests/anamnese.test.js
PASS tests/agendamento.test.js
PASS tests/odontograma.test.js
PASS tests/aluno.test.js
PASS tests/indice_sangramento.test.js
PASS tests/indice_biofilme.test.js
PASS tests/cpos_ampliado.test.js
PASS tests/exame.test.js
PASS tests/cidade-estado.test.js
PASS tests/authRoute.test.js

Test Suites: 16 passed, 16 total
Tests:      133 passed, 133 total
Snapshots: 0 total
Time:       6.937 s

tests
├── seeders
│   └── JS test-seeder.js
├── JS agendamento.test.js
├── JS aluno.test.js
├── JS anamnese.test.js
├── JS auth.test.js
├── JS authRoute.test.js
├── JS cidade-estado.test.js
├── JS cpos_ampliado.test.js
├── JS diario_dieta-refeicao.test.js
├── JS exame.test.js
├── JS indice_biofilme.test.js
├── JS indice_sangramento.test.js
├── JS odontograma.test.js
├── JS paciente.test.js
├── JS plano_tratamento-item.test.js
├── JS professor.test.js
└── JS secretaria.test.js

```

Figura 6 – Resultados dos testes automatizados e estrutura da suíte de testes

Foi adotada novamente a utilização de *seeders* específicos para o ambiente de testes, com o objetivo de inicializar o sistema com dados essenciais à execução dos cenários testados. Esses *seeders* permitem a criação automática de usuários com diferentes perfis e permissões, bem como de registros básicos necessários para a realização das operações e validação das regras de negócio.

A presença desses dados iniciais possibilitou a simulação de situações reais de uso do sistema, garantindo que os testes fossem executados em um contexto consistente e controlado. Dessa forma, foi possível validar não apenas o funcionamento isolado das funcionalidades, mas também o comportamento do sistema em fluxos completos de operações.

Os testes contemplaram todos os principais modelos do sistema, abrangendo uma ampla variedade de cenários, incluindo:

- Operações realizadas com sucesso.
- Tentativas de inserção ou atualização com dados inválidos ou inconsistentes.
- Acessos sem permissão adequada.
- Comportamentos esperados em situações de erro.
- Validação de regras de negócio.

Esse conjunto de testes contribuiu para assegurar a robustez da aplicação, reduzindo a probabilidade de falhas em produção e aumentando a confiabilidade do sistema. A Figura 6 apresenta o resultado da execução dos testes automatizados, evidenciando a aprovação de todas as suítes de teste definidas para o projeto.

Conforme mencionado anteriormente, o desenvolvimento do sistema seguiu os princípios do método *Test-Driven Development (TDD)*, no qual os testes são definidos antes ou em paralelo à implementação das funcionalidades. Essa abordagem contribuiu para a identificação precoce de erros, para a melhoria da qualidade do código e para a garantia de que as funcionalidades atendessem aos requisitos especificados.

3.7 VERIFICAÇÃO DE REQUISITOS E AVALIAÇÃO COM USUÁRIOS

Para assegurar que o SISCAR cumpre os objetivos estabelecidos no início do projeto, foram realizadas atividades de validação ao longo do desenvolvimento do sistema. A verificação dos requisitos funcionais foi conduzida por meio da implementação incremental das funcionalidades e da execução de testes manuais baseados em cenários de uso que simulavam as atividades reais realizadas na clínica. Para cada requisito funcional definido, foi verificado se o sistema permitia executar a ação correspondente, como cadastro, edição,

visualização ou remoção de registros, bem como o correto funcionamento das permissões associadas a cada perfil de usuário.

No caso dos requisitos não funcionais, a verificação ocorreu principalmente por meio da análise da arquitetura implementada e de testes práticos durante o uso do sistema. Aspectos como controle de acesso foram avaliados por meio da autenticação com diferentes perfis de usuário, garantindo que cada papel tivesse acesso apenas às funcionalidades permitidas. Já requisitos relacionados à usabilidade e organização da interface foram analisados durante a utilização do sistema e refinados ao longo do desenvolvimento.

Além disso, protótipos e versões parciais do sistema foram apresentados a integrantes da clínica durante reuniões de acompanhamento do projeto. Nessas ocasiões, foram coletados *feedbacks* sobre a organização das telas, fluxo de navegação e adequação das funcionalidades às rotinas da clínica. Essas interações permitiram ajustar detalhes de interface e confirmar que as funcionalidades implementadas atendiam às necessidades levantadas inicialmente.

Esse processo iterativo de desenvolvimento, validação e refinamento contribuiu para assegurar que os requisitos definidos fossem efetivamente contemplados na implementação do sistema SISCAR.

4 RESULTADOS

Neste capítulo, apresentam-se os resultados tangíveis obtidos a partir do desenvolvimento do sistema SISCAR. Por meio da exposição das interfaces gráficas e da descrição detalhada de suas funcionalidades, demonstra-se como os requisitos levantados e as escolhas tecnológicas foram materializados em uma solução funcional e multiplataforma. As seções a seguir detalham o fluxo de uso da aplicação, abrangendo desde o controle de acesso e gestão de usuários até os módulos específicos de acompanhamento clínico e exames odontológicos, evidenciando a adaptação do sistema para diferentes perfis de usuário e dispositivos (*desktop* e *mobile*).

4.1 TELAS DO APLICATIVO

4.1.1 Tela de Login

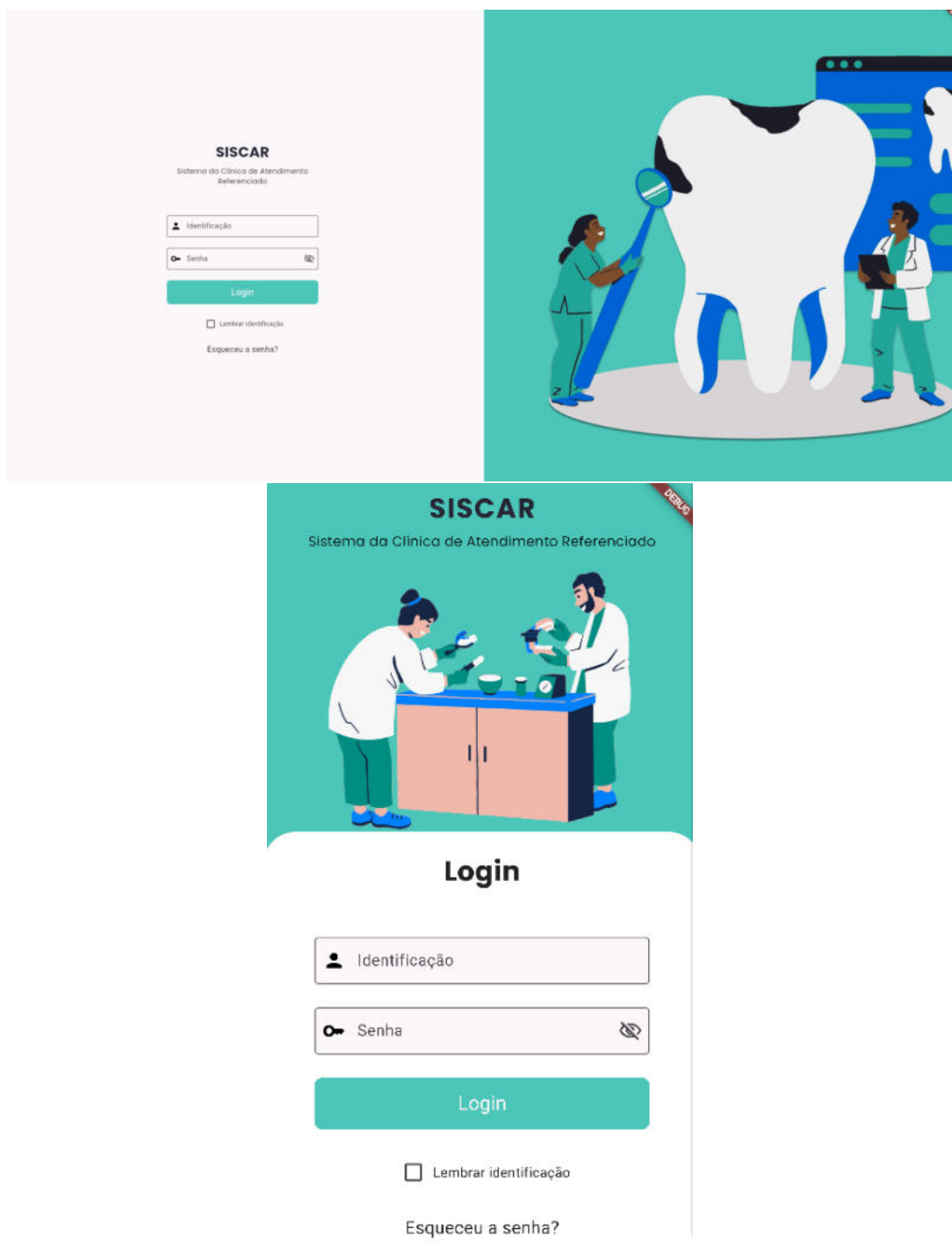


Figura 7 – Tela de *login* na versão *desktop* e *mobile*

A Tela de *Login* (Figura 7) do sistema SISCAR corresponde ao ponto de entrada da aplicação, sendo responsável pelo processo de autenticação dos usuários. Nessa tela, o usuário deve informar suas credenciais de acesso (CPF e senha) para que o sistema

possa validar sua identidade e conceder acesso às funcionalidades de acordo com o perfil associado (professor, aluno ou secretária). Ao lado do campo de senha há um botão que permite a visualização do que foi digitado.

Além dos campos de autenticação, a interface apresenta recursos auxiliares, como a opção de lembrar a identificação do usuário que é uma *checkbox* que ao ser clicada salva as informações para depois quando acessar a tela novamente já ter a identificação preenchida. Além disso também há uma opção de “esqueceu a senha”(ainda não implementado). Mas que futuramente irá redirecionar o usuário para uma tela de recuperação de senha.

4.1.2 Tela Inicial

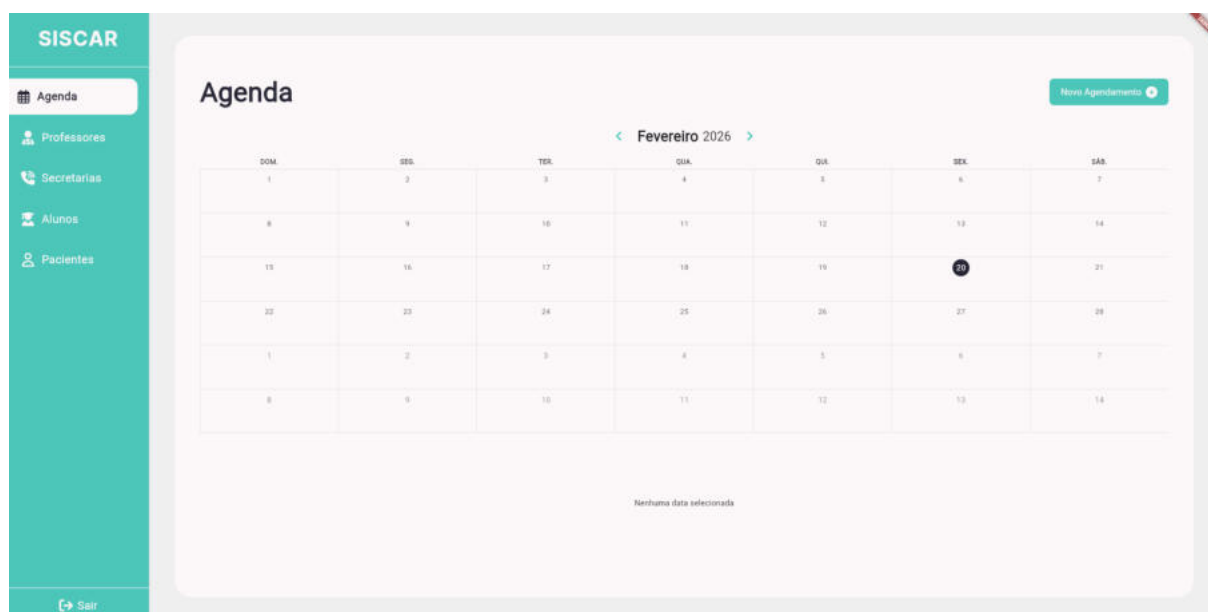


Figura 8 – Tela inicial na versão *desktop*

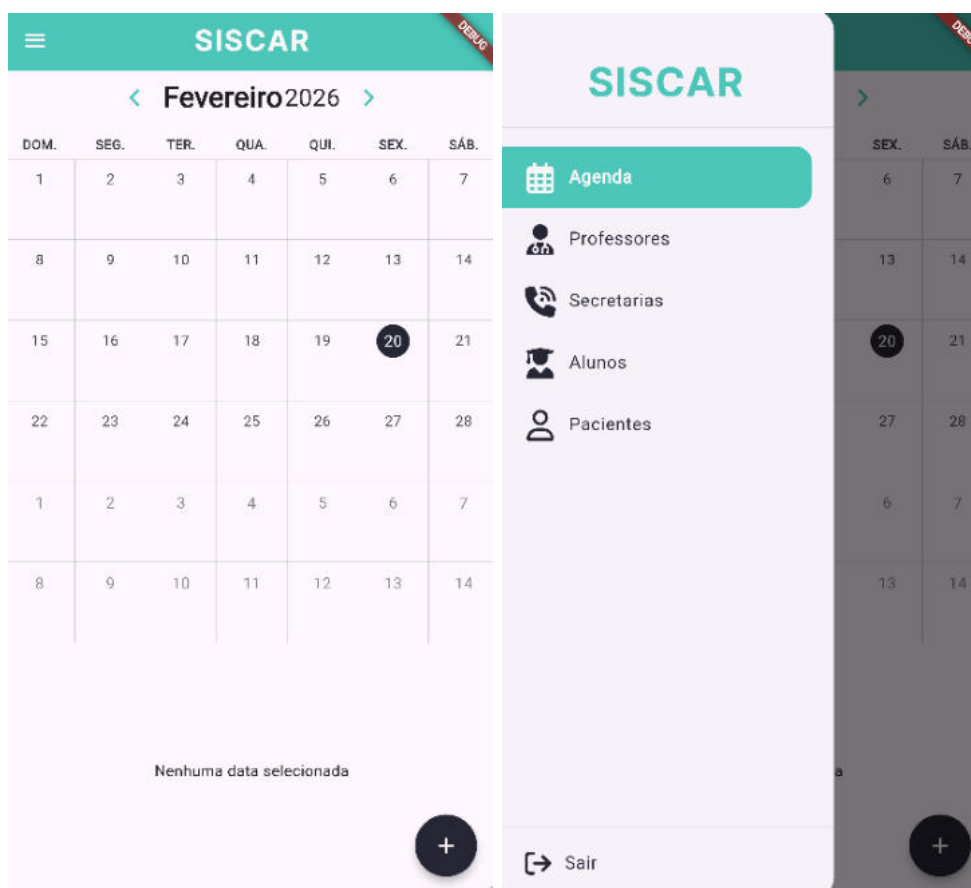


Figura 9 – Tela inicial na versão *mobile*

Após a autenticação, o usuário é direcionado à Tela Inicial do sistema, cuja funcionalidade central é a Agenda. Essa tela apresenta um calendário mensal, no qual é possível visualizar os dias do mês corrente, com destaque visual para a data atual, facilitando a orientação temporal e o planejamento dos atendimentos.

O calendário permite a navegação entre meses e funciona como o principal ponto de acesso para o gerenciamento de agendamentos da clínica. Caso tenha um atendimento no dia, um círculo aparecerá abaixo da data. Ao selecionar uma data específica, o sistema exibe as informações correspondentes aos atendimentos programados abaixo do calendário onde no momento está escrito “Sem consultas para esta data”, possibilitando uma visão organizada da rotina clínica.

Além do calendário, a interface conta com um *menu* lateral de navegação, responsável por dar acesso às demais funcionalidades do sistema. Esse *menu* é dinâmico e adaptado de acordo com o perfil do usuário autenticado, reforçando as regras de controle de acesso definidas na modelagem do sistema. Usuários do tipo secretária possuem acesso apenas à agenda, enquanto alunos visualizam a agenda e a área de pacientes, porém sem permissão para realizar novos agendamentos. Já o perfil de professor, ilustrado nas Figuras 8 e 9, possui acesso completo às funcionalidades do sistema, incluindo o cadastro de novos agendamentos, gerenciamento de usuários e acesso aos registros clínicos.

4.1.3 Tela de Agendamento

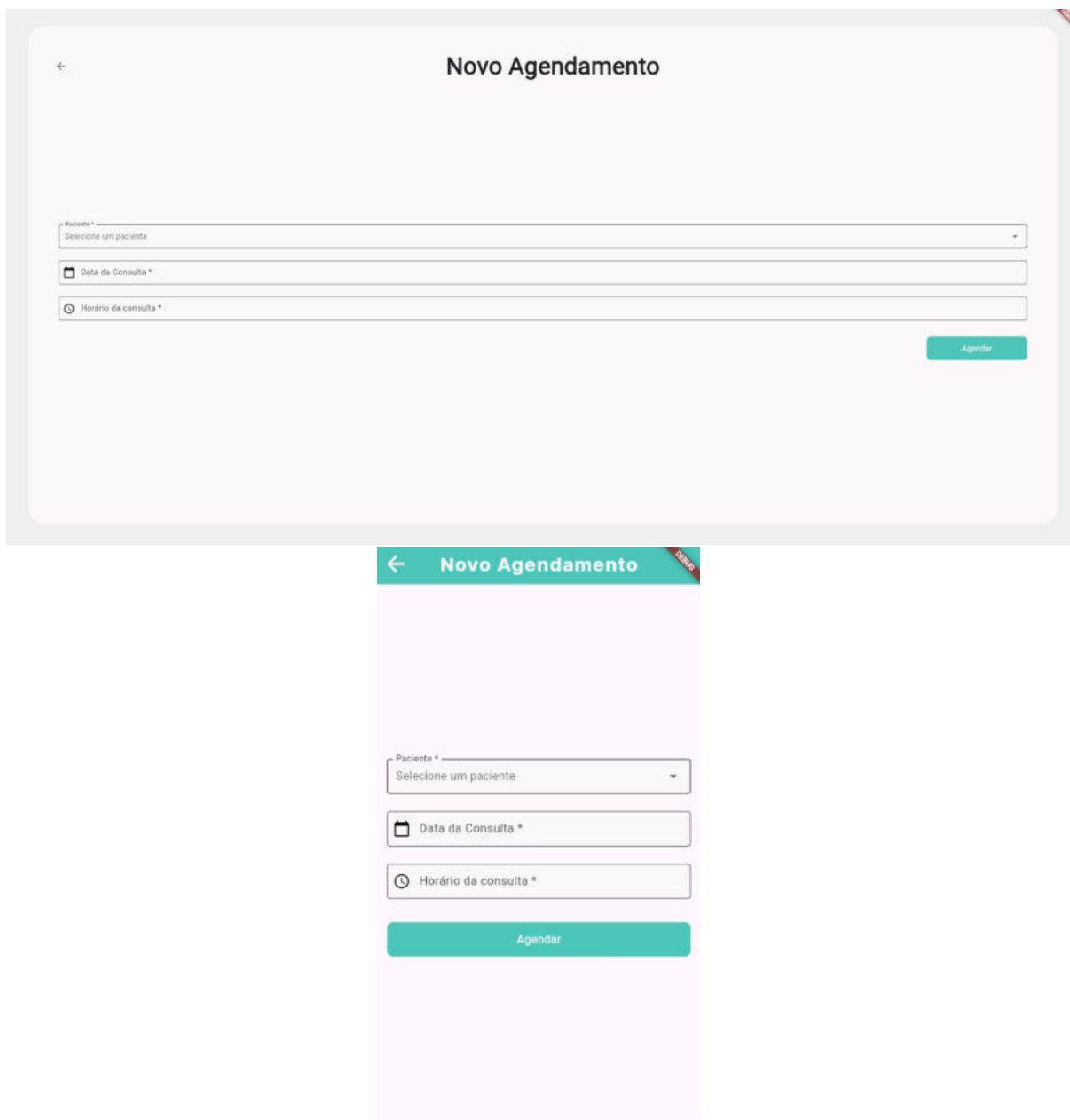


Figura 10 – Tela de agendamento na versão *desktop* e *mobile*

A Tela de Agendamento, ilustrado na Figura 10, é acessada a partir da Tela Inicial por meio do botão “Novo Agendamento”. Ao selecionar essa opção, o usuário é direcionado para uma interface dedicada ao cadastro de uma nova consulta.

Essa tela é composta por campos essenciais para a criação do agendamento. O paciente é selecionado por meio de um campo do tipo *dropdown*, que lista todos os pacientes previamente cadastrados no sistema, garantindo a vinculação correta do atendimento. A data da consulta é definida através de um componente do tipo *datepicker*, facilitando a seleção visual do dia desejado, enquanto o horário da consulta é informado por meio de

um *timepicker*, assegurando precisão no registro do horário do atendimento.

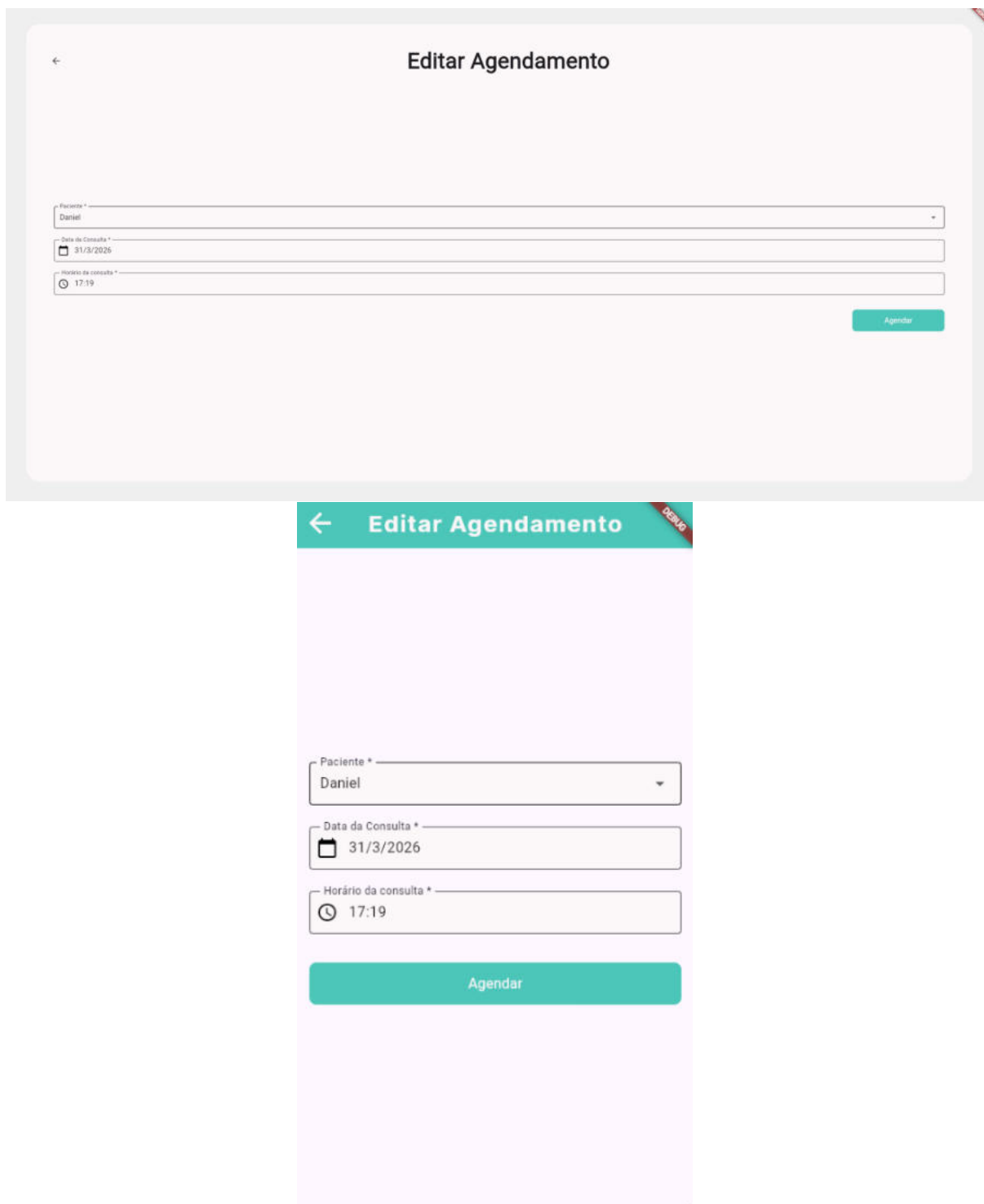


Figura 11 – Tela de edição de agendamento na versão *desktop* e *mobile*

Além do cadastro de novos agendamentos, essa mesma tela também é reutilizada para a edição de agendamentos existentes, ilustrado na Figura 11. Quando o usuário seleciona um dia que possui consultas já cadastradas e clica em um dos agendamentos listados abaixo do calendário, o sistema direciona automaticamente para a Tela de Agendamento

com os campos previamente preenchidos. Nesse contexto, o usuário pode apenas atualizar as informações necessárias, mantendo a consistência dos dados e evitando retrabalho.

4.1.4 Tela de Professores – Listagem

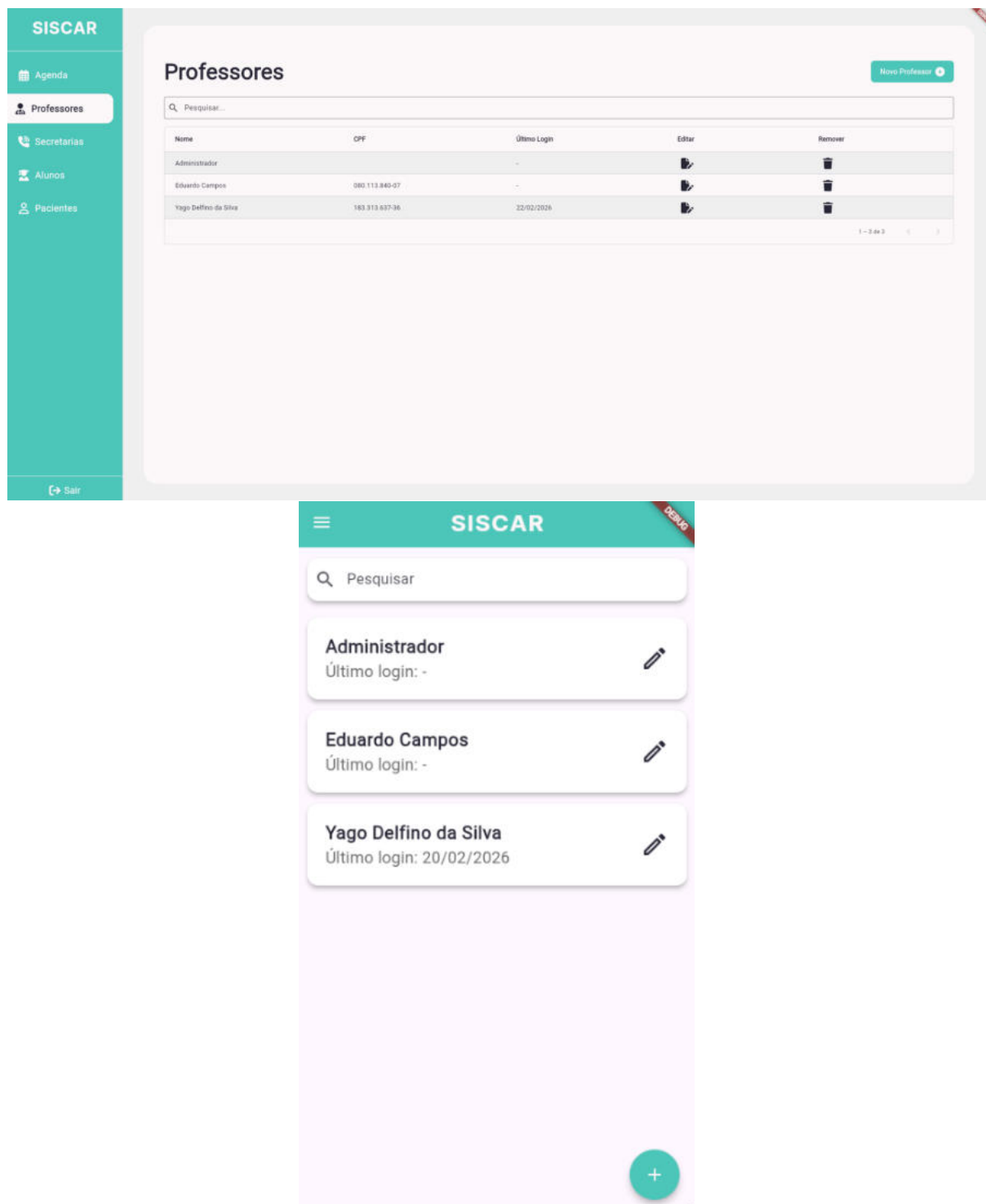


Figura 12 – Tela da lista de professores na versão *desktop* e *mobile*

A Tela de Professores começa com uma lista de todos os professores que estão cadastrados no sistema. Na interface de listagem, são apresentados os principais dados de cada professor cadastrado, como nome, CPF e a data do último *login* no sistema. Essas informações permitem à administração da clínica acompanhar a atividade dos usuários e identificar perfis ativos ou inativos. Além disso, a tela conta com um campo de busca, que possibilita a filtragem dos registros de forma rápida e eficiente, facilitando a localização de um professor específico mesmo em cenários com grande volume de dados.

4.1.5 Tela de Professores – Cadastro e Edição

The image displays two versions of the 'Novo Professor' registration form. The top version is the desktop interface, featuring a light gray background and a white form area. The form is titled 'Novo Professor' and includes a back arrow on the left. It contains six input fields: 'Nome Civil', 'Nome Social', 'CPF', 'E-mail', 'Senha', and 'Confirmar Senha'. Each field has a corresponding icon (person, social media, document, envelope, and lock). A green 'Salvar' button is positioned at the bottom right of the form. The bottom version is the mobile interface, showing the same form on a vertical screen. It has a teal header with a back arrow and the title 'Novo Professor'. The input fields are stacked vertically, and the green 'Salvar' button is centered at the bottom.

Figura 13 – Tela de cadastro de professor na versão *desktop* e *mobile*

Da tela de listagem, ao clicar no botão “Novo Professor” na versão *desktop* (Figura 21) ou no botão flutuante verde com o símbolo “+” na versão *mobile* (Figura 21), o sistema redireciona o usuário para a tela de cadastro de um novo professor.

Nesta tela, são apresentados os campos necessários para a criação do usuário professor no sistema, incluindo nome civil, nome social, CPF, e-mail, senha e confirmação de senha. A disposição dos campos foi pensada de forma a manter a consistência visual entre as versões *web* e *mobile*, garantindo uma experiência de uso semelhante independentemente da plataforma utilizada.

Após o preenchimento correto das informações, ao acionar o botão "Salvar", os dados são enviados ao *backend* para persistência no banco de dados. Caso o cadastro seja realizado com sucesso, o sistema retorna para a tela de listagem de professores, onde o novo registro passa a ser exibido imediatamente. Em situações de erro, como tentativa de cadastro com CPF ou e-mail já existentes, o sistema informa o usuário por meio de mensagens de *feedback* adequadas. Ou caso o CPF ou e-mail sejam inválidos, os próprios campos avisam, ficando vermelhos.

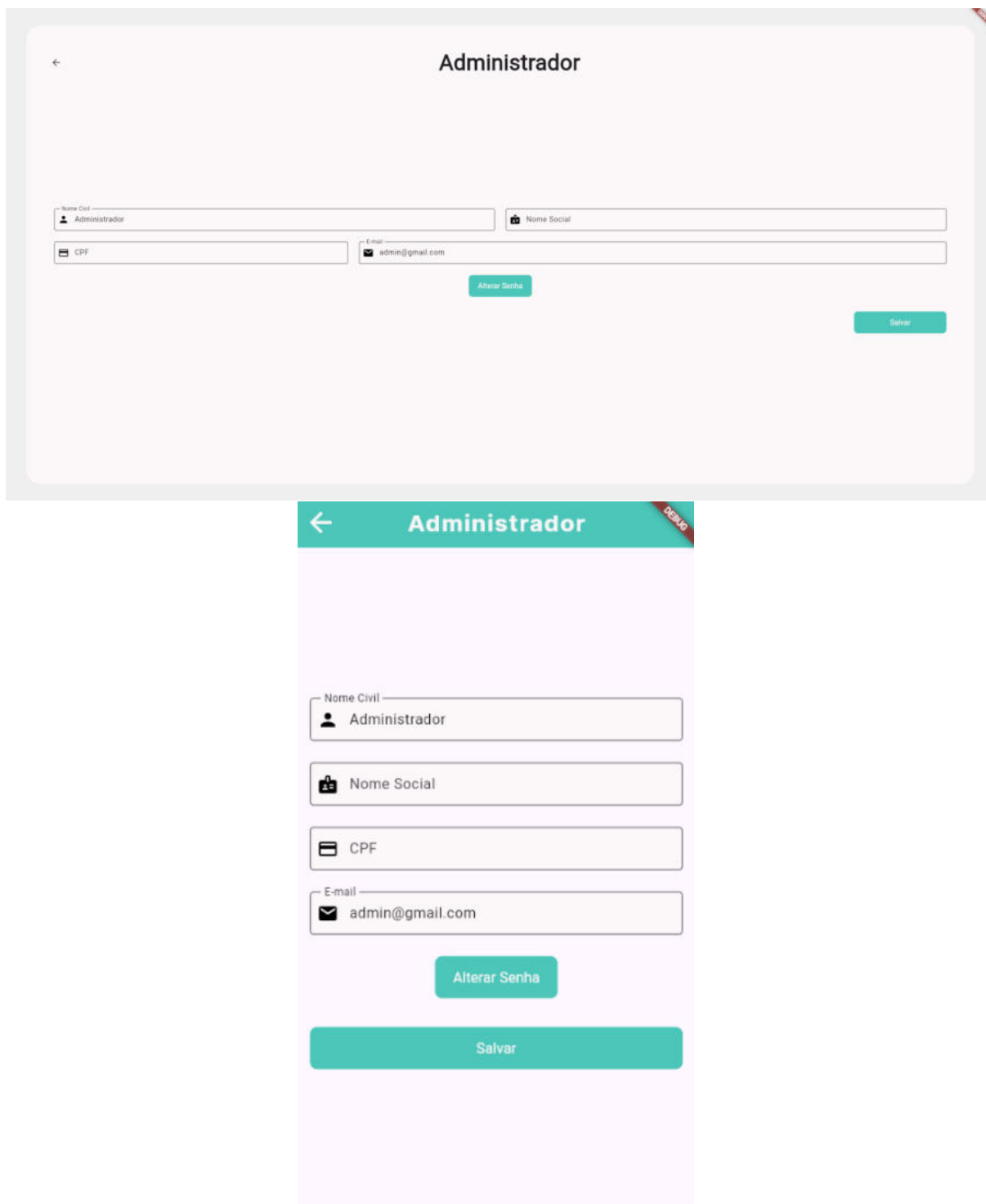


Figura 14 – Tela de edição de professor na versão *desktop* e *mobile*

A mesma tela de cadastro é reutilizada na edição, todavia, o acesso é diferente. Quando o usuário seleciona um professor já cadastrado na listagem, por meio do ícone de edição, o sistema redireciona para a mesma interface de formulário, porém com os campos previamente preenchidos com os dados existentes no banco de dados.

O campo de senha, por sua vez, não é exibido diretamente para edição, sendo substituído por um botão específico de “Alterar Senha”. Quando o botão é clicado abre um

modal com os campos “senha” e “confirmar senha”.

4.1.6 Tela de Alunos – Listagem

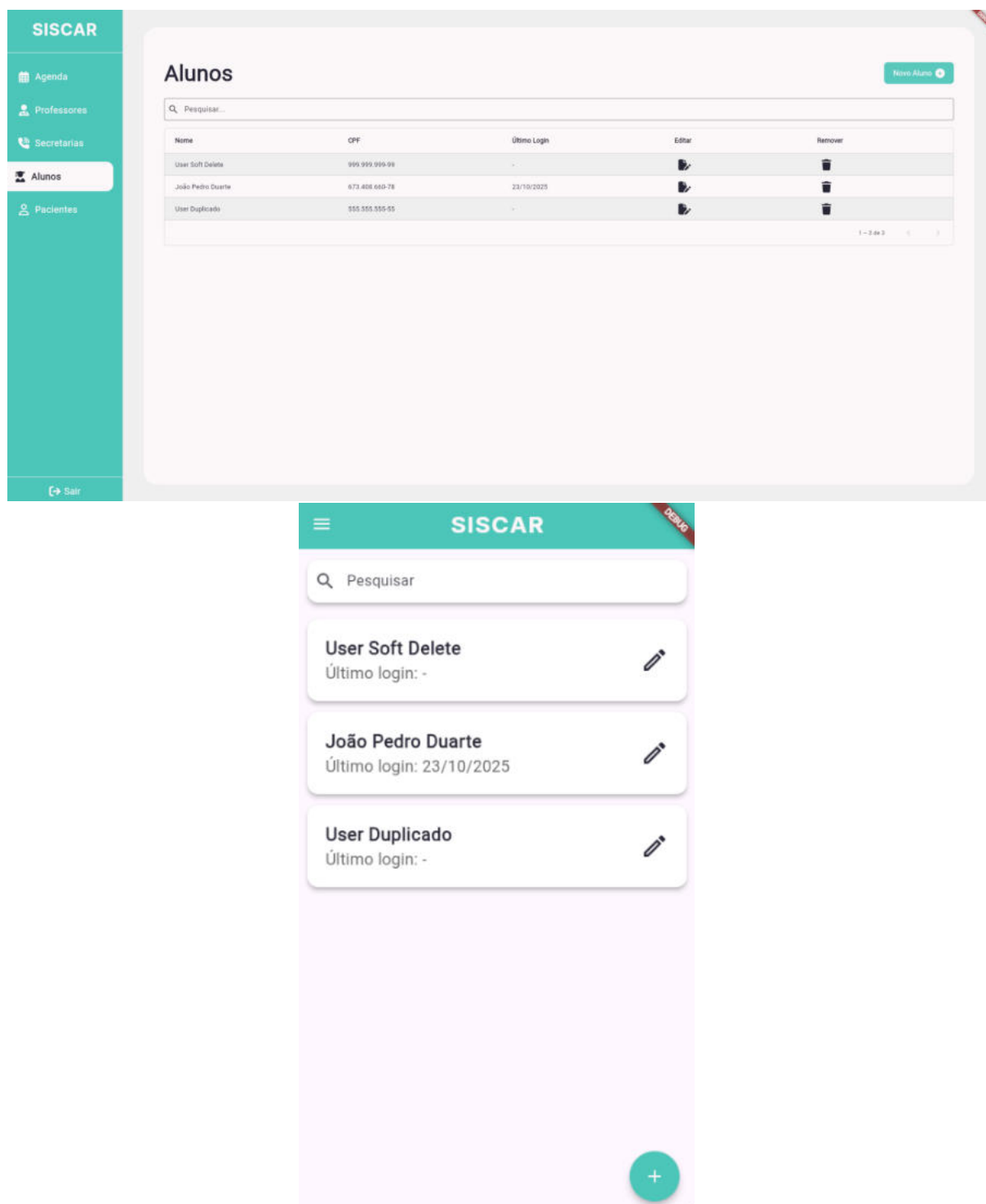


Figura 15 – Tela da lista de alunos na versão *desktop* e *mobile*

A Tela de Alunos segue a mesma estrutura visual e organizacional da Tela de Professores, mantendo a consistência da interface do sistema. Nessa tela, é apresentada a

listagem dos alunos cadastrados, exibindo informações essenciais para identificação, como nome, CPF e data do último acesso ao sistema.

Assim como na listagem de professores, a interface disponibiliza um campo de busca que permite a filtragem dinâmica dos registros, facilitando a localização de um aluno específico mesmo em cenários com grande volume de dados. A adoção de uma interface padronizada contribui para a usabilidade do sistema, reduzindo a curva de aprendizado dos usuários e promovendo uma experiência uniforme entre os diferentes perfis.

4.1.7 Tela de Alunos – Cadastro e Edição

The image displays two versions of the 'Novo Aluno' (New Student) registration form. The top screenshot is the desktop version, showing a form with six input fields: 'Nome Civil', 'Nome Social', 'CPF', 'E-mail', 'Senha', and 'Confirmar Senha'. A 'Salvar' button is located at the bottom right. The bottom screenshot is the mobile version, showing the same form with the fields stacked vertically and a 'Salvar' button at the bottom.

Figura 16 – Tela de cadastro de aluno na versão *desktop* e *mobile*

As telas de cadastro e edição de alunos reutilizam o mesmo padrão visual e funcional adotado no cadastro de professores. O formulário apresenta os campos necessários para o registro do aluno no sistema, garantindo consistência entre as versões *web* e *mobile* da aplicação.

O processo de cadastro é iniciado a partir da tela de listagem, por meio do botão de criação de novo registro. Após o preenchimento correto dos campos obrigatórios, os dados são enviados ao *backend* para validação e persistência no banco de dados. Em caso de sucesso, o sistema redireciona o usuário para a listagem de alunos atualizada e em situações de erro, mensagens de *feedback* são exibidas, indicando inconsistências ou dados inválidos.

The image displays two versions of a web form for editing a student's profile. The top screenshot is the desktop version, showing a form with four input fields: 'Nome Civil' (filled with 'João Pedro Duarte'), 'Nome Social', 'CPF' (filled with '673.408.660-78'), and 'E-mail' (filled with 'jp_dudu@bol.com.br'). Below the fields are two buttons: 'Alterar Senha' and 'Salvar'. The bottom screenshot is the mobile version, showing the same form with a teal header bar containing a back arrow and the name 'João Pedro Duarte'. The input fields and buttons are stacked vertically, with 'Alterar Senha' and 'Salvar' buttons being wider and more prominent.

Figura 17 – Tela de edição de aluno na versão *desktop* e *mobile*

A funcionalidade de edição utiliza a mesma interface do cadastro, com os campos previamente preenchidos com as informações existentes, permitindo a atualização dos dados de forma simples e segura.

4.1.8 Tela de Secretárias – Listagem

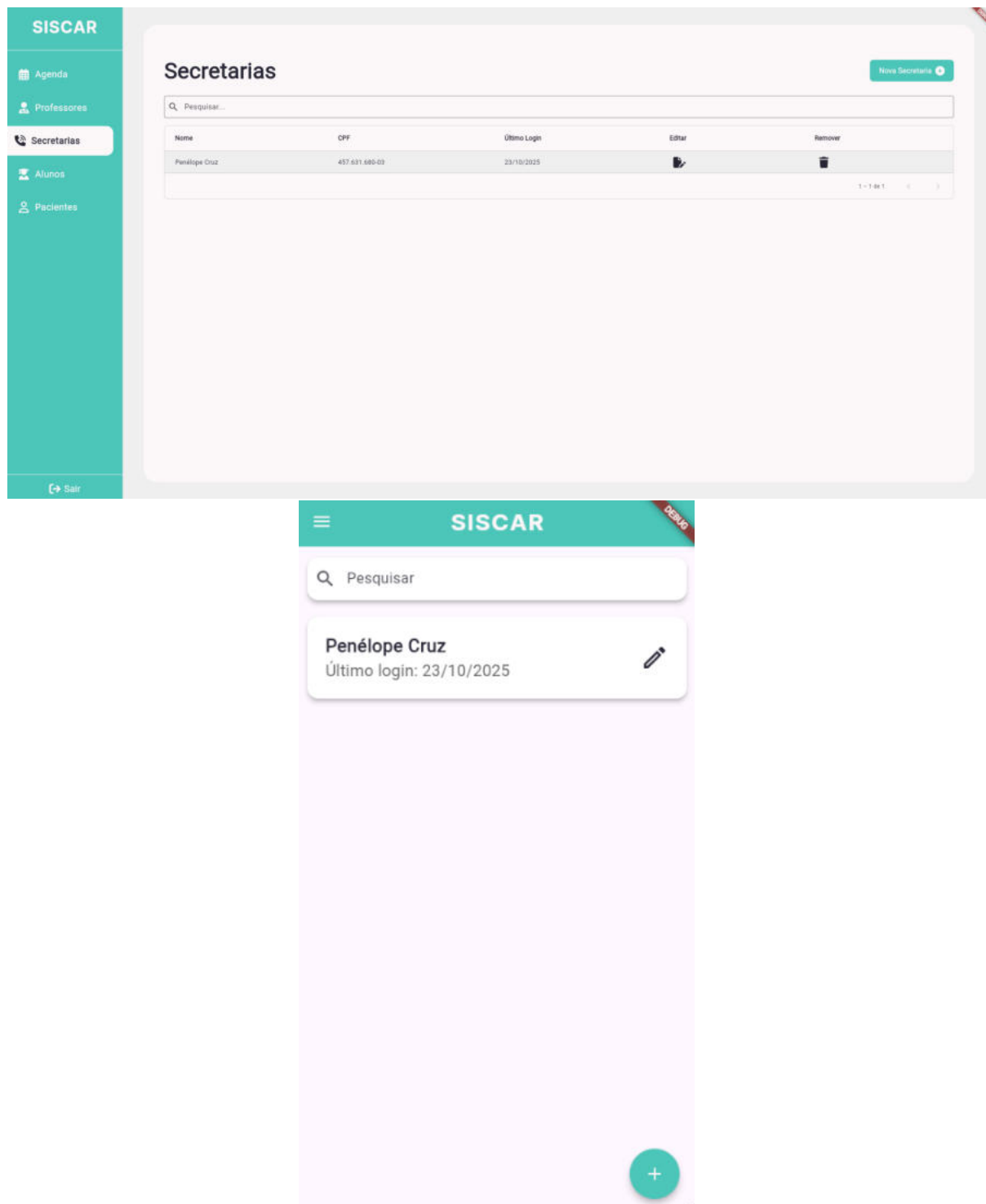


Figura 18 – Tela da lista de secretárias na versão *desktop* e *mobile*

A Tela de Secretárias também segue o padrão visual adotado nas telas de Professores e Alunos, reforçando a padronização da interface do sistema. Nessa tela, são listadas as secretárias cadastradas, com a apresentação dos principais dados de identificação necessários para a administração do sistema.

Apesar da similaridade visual, as funcionalidades disponíveis para este perfil são definidas de acordo com as permissões associadas ao papel de secretária, restringindo o acesso a recursos sensíveis do sistema, como prontuários clínicos e informações acadêmicas.

4.1.9 Tela de Secretárias – Cadastro e Edição

A imagem mostra duas versões da interface de usuário para o cadastro de uma nova secretária. A versão superior é a versão desktop, com o título "Nova Secretaria" e um ícone de seta para trás no canto superior esquerdo. O formulário contém campos para "Nome Civil", "Nome Social", "CPF", "E-mail", "Senha" e "Confirmar Senha", cada um com um ícone de campo correspondente. Um botão "Salvar" em verde está no canto inferior direito. A versão inferior é a versão mobile, com o mesmo título e ícone de seta para trás no topo. O formulário é adaptado para o formato vertical, com os campos "Nome Civil", "Nome Social", "CPF", "E-mail", "Senha" e "Confirmar Senha" empilhados verticalmente. Um botão "Salvar" em verde está no final da lista de campos.

Figura 19 – Tela de cadastro de secretária na versão *desktop* e *mobile*

As telas de cadastro e edição de secretárias utilizam a mesma estrutura de formulário aplicada aos demais perfis de usuário do sistema. Essa abordagem permite a reutilização de componentes e garante uniformidade visual entre os diferentes módulos da aplicação.

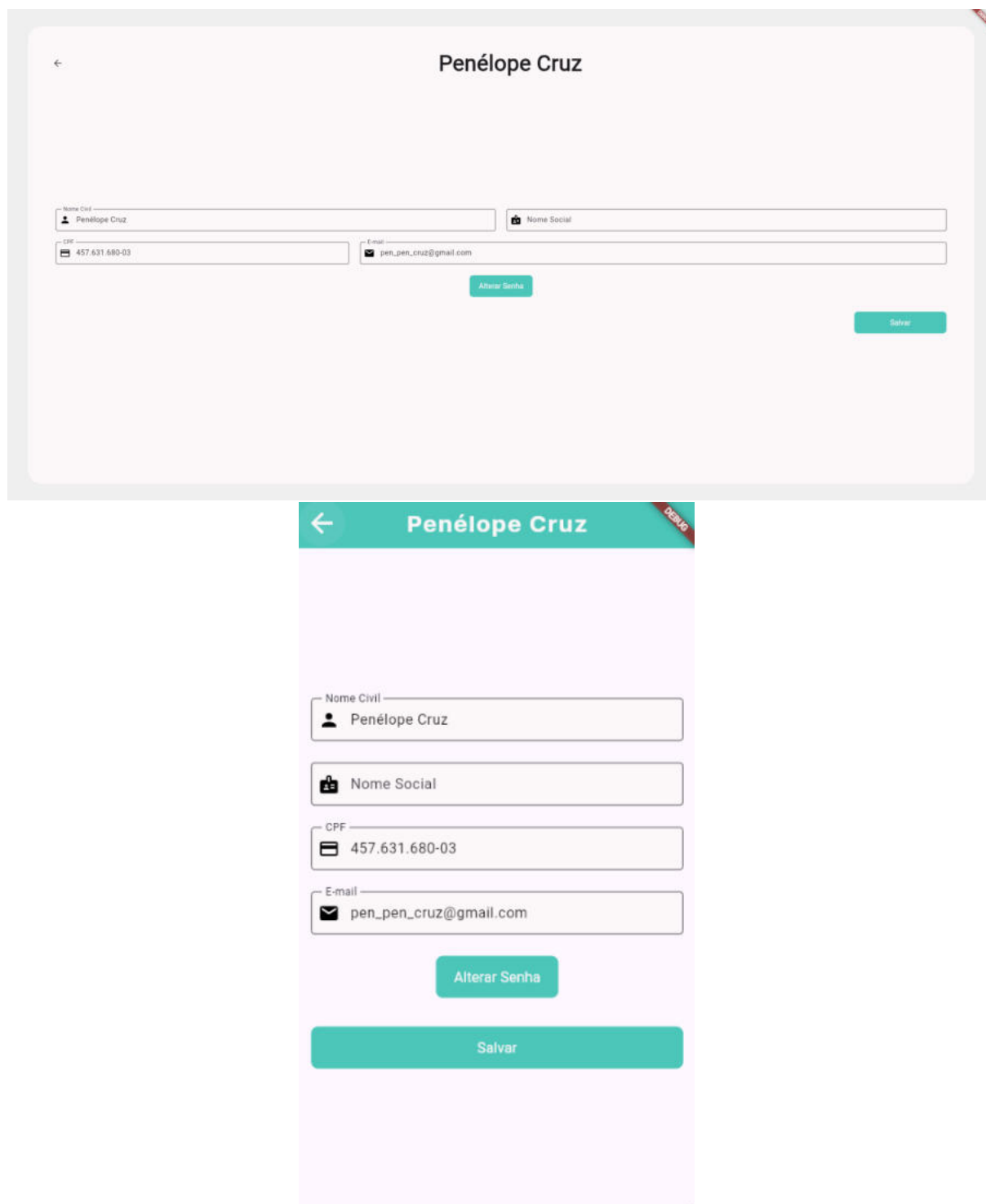


Figura 20 – Tela de edição de secretária na versão *desktop* e *mobile*

O fluxo de criação e edição segue o mesmo padrão já descrito anteriormente: a partir da listagem, o usuário autorizado pode cadastrar uma nova secretária ou editar um registro

existente. O sistema realiza as validações necessárias antes da persistência dos dados, exibindo mensagens de *feedback* em casos de sucesso ou erro.

4.1.10 Tela de Pacientes – Listagem

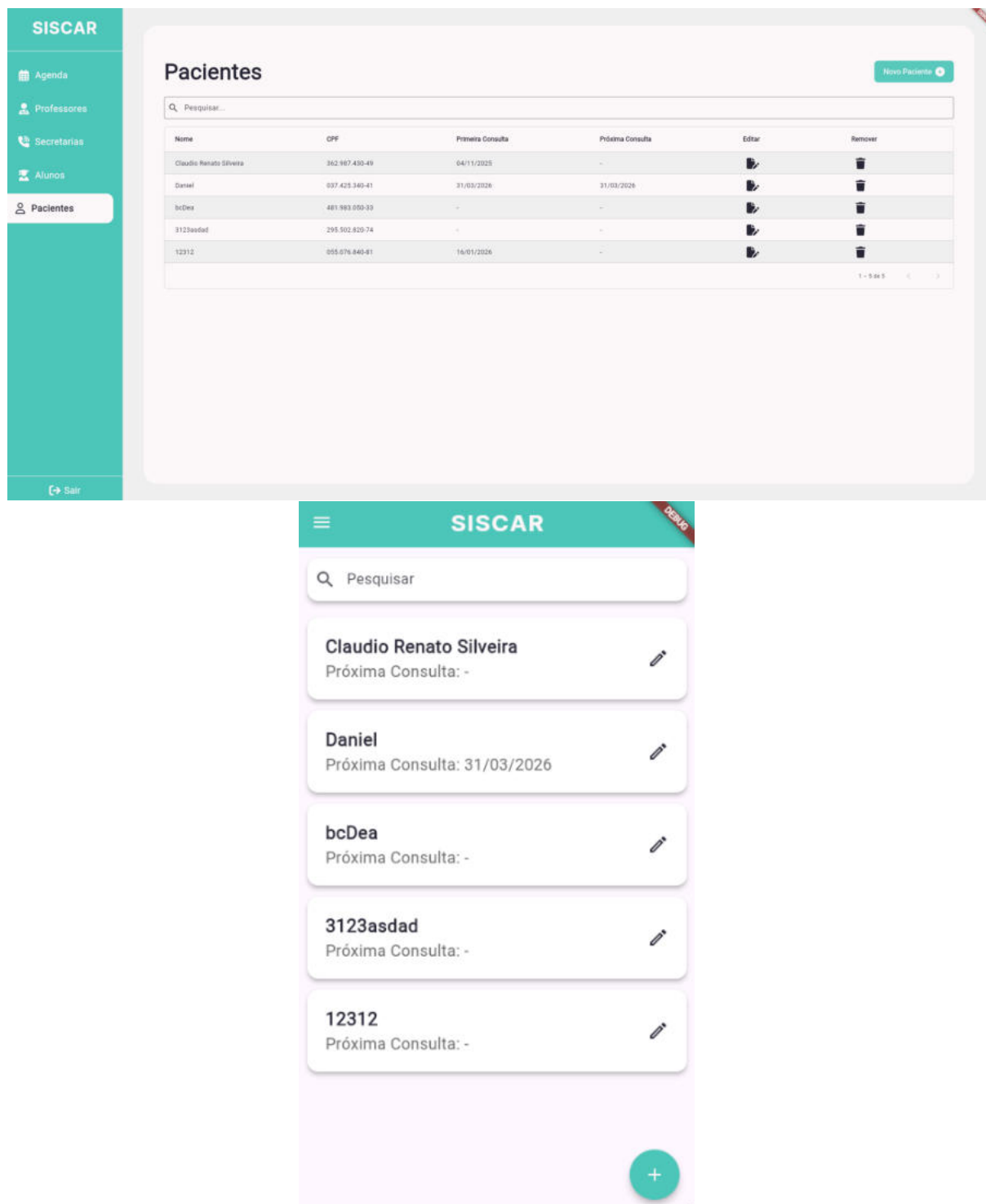


Figura 21 – Tela da lista de pacientes na versão *desktop* e *mobile*

A Tela de Pacientes apresenta a listagem de todos os pacientes cadastrados no sistema, mantendo o mesmo padrão visual e estrutural adotado nas telas de gerenciamento de usuários administrativos do sistema, como professores, alunos e secretárias, garantindo consistência na experiência de uso.

Na versão *desktop* (Figura 21), a listagem exibe informações relevantes para o acompanhamento clínico, incluindo o nome do paciente, CPF, data da primeira consulta realizada e a data da próxima consulta agendada. Esses dados permitem uma visualização rápida da situação clínica de cada paciente e facilitam a organização do fluxo de atendimentos.

Já na versão *mobile* (Figura 21), devido às limitações de espaço de tela, são apresentados apenas o nome do paciente e a data da próxima consulta, priorizando as informações mais relevantes para o uso em dispositivos móveis. A tela também conta com um campo de busca, que possibilita a filtragem dos pacientes de forma dinâmica e eficiente.

4.1.11 Tela de Paciente – Menu

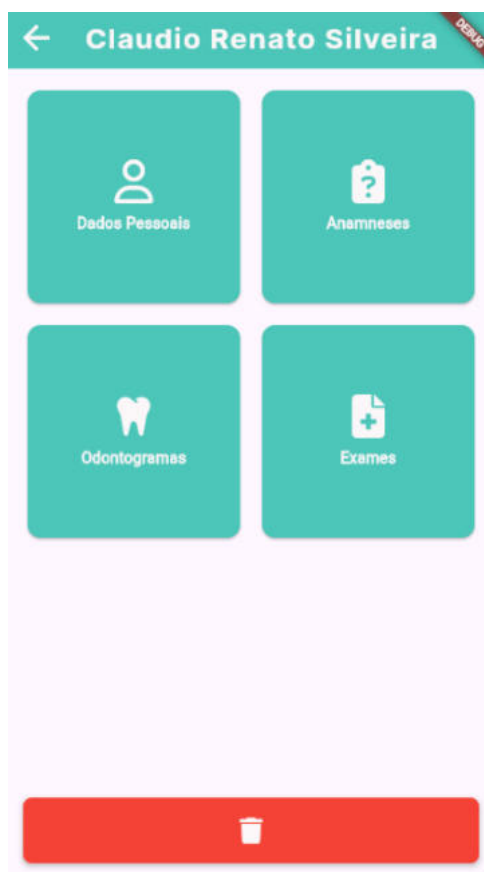


Figura 22 – Tela de *menu* de um paciente no *mobile*

Após a seleção de um paciente na versão *mobile* do sistema, o usuário é direcionado para a Tela de *Menu* do Paciente. Essa tela funciona como um ponto central de navegação da ficha clínica, apresentando de forma organizada os principais módulos relacionados ao atendimento odontológico.

O *menu* é composto por quatro botões principais, cada um representando uma seção específica da ficha do paciente: Dados Pessoais, Anamneses, Odontogramas e Exames. A disposição dos botões em formato de cartões facilita a visualização e o acesso rápido às informações, contribuindo para uma navegação intuitiva e eficiente, especialmente em dispositivos móveis.

Além disso, a tela conta com um botão de exclusão do paciente, posicionado de forma destacada na parte inferior da interface. Essa ação é restrita a usuários com permissão adequada e permite a remoção do registro do paciente do sistema, reforçando a importância do controle de acesso e da segurança das informações clínicas.

4.1.12 Tela de Paciente – Dados Pessoais

The image displays two versions of the 'Novo Paciente' (New Patient) form. The top version is the desktop interface, which is a multi-step process. It features a progress bar at the top with six steps: 1. Dados Pessoais (highlighted), 2. Anamnese, 3. Odontograma, 4. Índices de Biofilme Visual, 5. Índices de Sangramento, and 6. CPDs Ampliados. The form includes fields for: Perfil (dropdown), Nome Civil, Nome Social, CPF, Celular, E-mail, Estado (dropdown), Cidade (dropdown), CEP, Bairro, Rua, Numero, Complemento, Data de Nascimento, Raça (dropdown), and Sexo (dropdown). A 'Próximo' button is located at the bottom right.

The bottom version is the mobile interface. It has a green header with a back arrow and the title 'Novo Paciente'. Below the header is a progress indicator showing '1 de 6' and 'Dados Pessoais'. The form fields are stacked vertically: Perfil (dropdown), Nome Civil, Nome Social, CPF, Celular, E-mail, Estado (dropdown), and Cidade (dropdown). A 'Próximo' button is at the bottom.

Figura 23 – Tela de dados pessoais de paciente na versão *desktop* e *mobile*

A Tela de Dados Pessoais corresponde à primeira etapa do fluxo de criação e edição de um paciente no sistema. Essa tela é acessada ao clicar no botão “Novo Paciente” na versão *desktop* (Figura 21) ou no botão flutuante verde com o símbolo “+” na versão *mobile* (Figura 21), redirecionando o usuário para um formulário estruturado em múltiplas etapas.

O formulário reúne as principais informações cadastrais do paciente, como perfil de atendimento, nome civil, nome social, CPF, telefone, e-mail, data de nascimento, raça, gênero e endereço completo. A organização dos campos foi planejada para facilitar o preenchimento e reduzir erros de digitação, mantendo consistência visual e funcional entre as versões *web* e *mobile*. Vale ressaltar também que a pedido do *stakeholder*, as opções de raça e gênero deveriam ser obrigatórias no preenchimento do formulário.

No topo da interface, o sistema apresenta um indicador de progresso dividido em seis etapas, que representam a sequência completa da ficha clínica do paciente. A primeira etapa corresponde aos Dados Pessoais, seguida por Anamnese, Odontograma, Índices de Biofilme Visível, Índices de Sangramento e CPOs Ampliados. Esse indicador permite ao usuário acompanhar claramente em qual etapa do processo se encontra.

Na versão *desktop* (Figura 23), todas as etapas são exibidas horizontalmente, permitindo uma visualização ampla do fluxo completo. Já na versão *mobile* (Figura 23), o indicador de progresso é apresentado de forma compacta, destacando a etapa atual e a numeração correspondente, garantindo melhor adaptação ao espaço reduzido da tela sem comprometer a compreensão do processo.

The figure displays two versions of a patient's personal data form. The top version is the desktop interface, featuring a header with the patient's name 'Claudio Renato Silveira' and a navigation menu with tabs for 'Dados Pessoais', 'Anamneses', 'Odontogramas', 'Exames', and 'Plano de Tratamento'. The form fields are arranged in a grid-like structure with labels and icons for each field. The bottom version is the mobile interface, which is a vertical stack of the same form fields, with a 'Salvar' button at the bottom.

Desktop Version Fields:

- Perfil: Vila residencial
- Nome Civil: Claudio Renato Silveira
- Nome Social: [Empty]
- CPF: 362.987.430-49
- Celular: (21) 99437-4581
- E-mail: claudio_renato@gmail.com
- Estado: Rio de Janeiro
- Cidade: Rio de Janeiro
- CEP: 21524-10
- Bairro: Pavuna
- Rua: Praça Niterói Fonseca
- Número: 164
- Complemento: [Empty]
- Data de Nascimento: 4/11/1984
- Raça: Branco
- Gênero: Homem Cis

Mobile Version Fields:

- Perfil: Vila residencial
- Nome Civil: Claudio Renato Silveira
- Nome Social: [Empty]
- CPF: 362.987.430-49
- Celular: (21) 99437-4581
- E-mail: claudio_renato@gmail.com
- Estado: Rio de Janeiro
- Cidade: Rio de Janeiro
- CEP: 21524-10
- Bairro: [Empty]

Figura 24 – Tela preenchida de dados pessoais de paciente na versão *desktop* e *mobile*

Após a criação do paciente, os dados pessoais podem ser consultados e editados a qualquer momento. Na versão *desktop* (Figura 24), isso é feito selecionando o paciente diretamente na listagem. Já na versão *mobile* (Figura 24), o usuário acessa a listagem de pacientes, seleciona o paciente desejado e, em seguida, escolhe a opção “Dados Pessoais” no *menu* do paciente, reutilizando a mesma interface de formulário, agora com os campos previamente preenchidos com as informações armazenadas no sistema.

4.1.13 Tela de Paciente – Anamnese

Novo Paciente

1 Dados Pessoais 2 **Anamnese** 3 Odontograma 4 Índices de Índice Visual 5 Índices de Sangramento 6 CPDs Ampliados

Informações Médicas Gerais

Está sob cuidados médicos? Sim Não

Toma algum medicamento? Sim Não

Tem alergia a alguma substância? Sim Não

Histórico Médico

Tem problemas cardíacos? Sim Não

Teve algum infarto, derrame ou AVC? Sim Não

Tem hipertensão? Sim Não

Como é sua pressão arterial?

Teve convulsões ou desmaios? Sim Não

Tem asma, bronquite ou rinite? Sim Não

Tem hepatite ou problemas hepáticos? Sim Não

Tem algum problema renal? Sim Não

Hábitos

Fuma cigarros? Sim Não

Consome bebidas alcoólicas? Sim Não

Usa drogas (entorpecentes, estimulantes, inibidores de apetite, etc.)? Sim Não

Se considera uma pessoa tensa ou nervosa? Sim Não

Pratica esportes radicais de impacto ou lutas? Sim Não

Está grávida? Sim Não

Existe mais alguma consideração, queixa ou doença que não tenha respondido e acha importante relatar?

Questões Odontológicas

Quantas vezes por dia escova os dentes?

Usa fio dental? Sim Não

Limpa a língua regularmente? Sim Não

Faz uso de enxaguatórios bucais? Sim Não

Novo Paciente

2 de 6 **Anamnese**

Informações Médicas Gerais

Está sob cuidados médicos? Sim Não

Toma algum medicamento? Sim Não

Tem alergia a alguma substância? Sim Não

Histórico Médico

Tem problemas cardíacos? Sim Não

Teve algum infarto, derrame ou AVC? Sim Não

Tem hipertensão? Sim Não

Como é sua pressão arterial?

Teve convulsões ou desmaios? Sim Não

Tem asma, bronquite ou rinite? Sim Não

Tem hepatite ou problemas hepáticos? Sim Não

Tem algum problema renal? Sim Não

Figura 25 – Tela de anamnese de paciente na versão *desktop* e *mobile*

A Tela de Anamnese corresponde à segunda etapa do processo de cadastro e acompanhamento do paciente no sistema. Nessa fase, são coletadas informações clínicas e comportamentais essenciais para o atendimento odontológico, permitindo um melhor en-

tendimento das condições gerais de saúde do paciente antes da realização de procedimentos.

O formulário de anamnese é organizado em seções temáticas, como informações médicas gerais, histórico médico, hábitos e questões odontológicas. As perguntas são apresentadas majoritariamente no formato de alternativas do tipo "Sim" ou "Não" em formato de *radio button*, com campos adicionais de texto exibidos quando necessário, como uso de medicamentos, pressão arterial ou outras observações clínicas importantes.

Assim como nas demais etapas do cadastro, a anamnese integra um fluxo guiado por um indicador de progresso localizado na parte superior da interface. Esse indicador apresenta visualmente todas as etapas da ficha clínica, numeradas sequencialmente, permitindo ao usuário identificar facilmente em qual fase do processo se encontra.

O sistema oferece flexibilidade na navegação entre as etapas. O usuário pode avançar para a próxima etapa por meio do botão "Próximo", retornar à etapa anterior utilizando o botão "Voltar" ou, caso necessário, utilizar o botão "Pular", permitindo que o preenchimento da ficha seja realizado de forma não linear, de acordo com a disponibilidade das informações no momento do atendimento. Etapas puladas podem ser revisitadas e completadas posteriormente, sem prejuízo ao cadastro do paciente.

O estado de cada etapa é indicado visualmente por cores no indicador de progresso. Etapas concluídas corretamente são marcadas em verde, sinalizando que todas as informações foram preenchidas e salvas. Etapas puladas são exibidas em amarelo, indicando pendência de preenchimento. Essa abordagem visual auxilia na orientação do usuário, reduz a possibilidade de inconsistências nos dados e contribui para uma experiência de uso mais clara e intuitiva.

Claudio Renato Silveira

Dados Pessoais | **Anamneses** | Odontogramas | Exames | Plano de Tratamento

Novo Anamnese

Anamnese de 16/01/2026
Anamnese de 04/11/2025

Informações Médicas Gerais

Está sob cuidados médicos? Sim Não
Especifique os cuidados médicos

Toma algum medicamento? Sim Não
Quais medicamentos?

Tem alergia a alguma substância? Sim Não
Quais substâncias?

Histórico Médico

Tem problemas cardíacos? Sim Não
Teve algum infarto, derrame ou AVC? Sim Não
Tem hipertensão? Sim Não
Como é sua pressão arterial?

Teve convulsões ou desmaios? Sim Não
Tem asma, bronquite ou rinite? Sim Não

Hábitos

Fuma cigarros? Sim Não
Quantos?

Consome bebidas alcoólicas? Sim Não
Qual frequência?

Usa drogas (entorpecentes, estimulantes, inibidores de apetite, etc.)? Sim Não
Se considera uma pessoa tensa ou nervosa? Sim Não
Pratica esportes radicais de impacto ou lutas? Sim Não
Está grávida? Sim Não
Existe mais alguma consideração, queixa ou doença que não tenha respondido e acha importante rel.?

Questões Odontológicas

Quantas vezes por dia escova os dentes?

Usa fio dental? Sim Não
Limpa a língua regularmente? Sim Não

Salvar Anamnese

Claudio Renato Silveira

Selecione a anamnese

+ Anamnese de 16/01/2026 Anamnese de 04/11/2025

Informações Médicas Gerais

Está sob cuidados médicos? Sim Não
Especifique os cuidados médicos

Toma algum medicamento? Sim Não
Quais medicamentos?

Tem alergia a alguma substância? Sim Não
Quais substâncias?

Histórico Médico

Tem problemas cardíacos? Sim Não
Teve algum infarto, derrame ou AVC? Sim Não
Tem hipertensão? Sim Não
Como é sua pressão arterial?

Teve convulsões ou desmaios? Sim Não

Figura 26 – Tela de lista de anamnese na versão *desktop* e *mobile*

Após a criação inicial do paciente, as anamneses podem ser consultadas e editadas a qualquer momento. Na versão *desktop* (Figura 26), o acesso ocorre diretamente ao selecionar o paciente na listagem e navegar até a aba de “Anamnese”. Na versão *mobile*, o usuário seleciona o paciente na listagem, acessa o *menu* do paciente e escolhe a opção “Anamnese”.

O sistema permite o cadastro de múltiplas anamneses para um mesmo paciente, consi-

derando que o paciente pode retornar à clínica após um determinado período e apresentar alterações em seu estado de saúde. Dessa forma, cada nova anamnese representa um novo registro temporal, preservando o histórico clínico e possibilitando a comparação entre atendimentos distintos, o que é essencial para garantir a segurança e a qualidade do acompanhamento odontológico.

Na versão *mobile*, a seleção das anamneses é apresentada na parte superior da tela, por meio de uma lista horizontal que exibe as datas dos registros existentes. Nesse contexto, também é disponibilizado um botão com o símbolo “+”, permitindo a criação de uma nova anamnese de forma rápida e intuitiva, facilitando o atendimento em situações em que o paciente retorna à clínica.

Já na versão *desktop* (Figura 26), a interface adota uma organização distinta, na qual a lista de anamneses é exibida em um painel lateral à esquerda da tela. Nesse painel, as anamneses são identificadas pela data de registro, possibilitando a seleção para visualização ou edição. No topo desse painel encontra-se o botão “Nova Anamnese”, que direciona o usuário para a criação de um novo registro.

4.1.14 Tela de Paciente – Odontograma

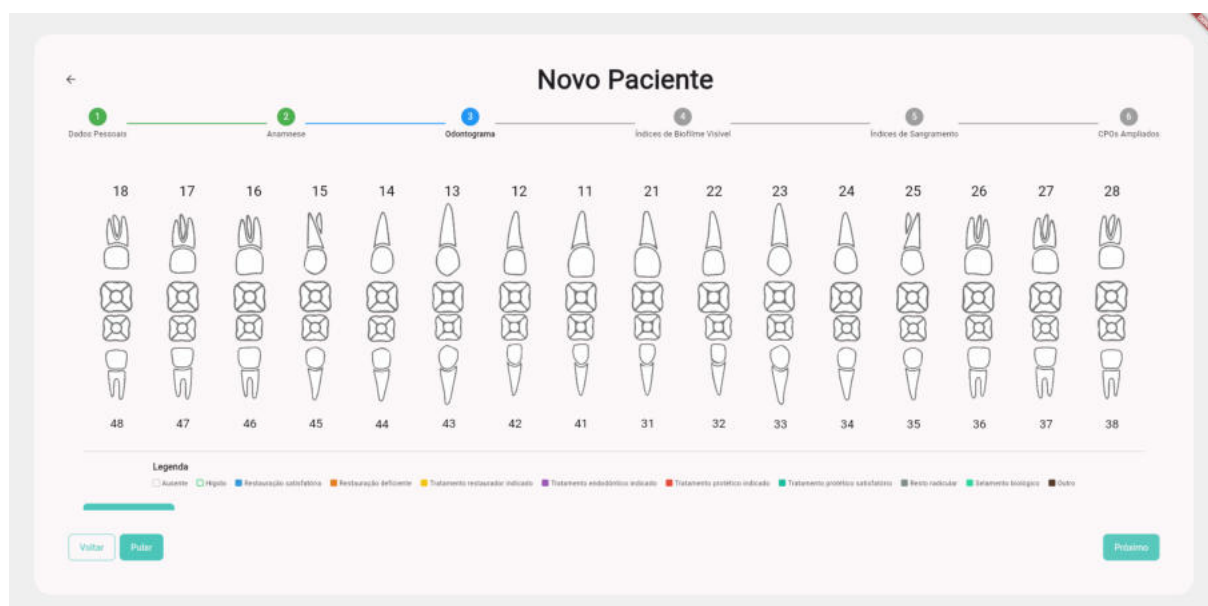


Figura 27 – Tela de odontograma de paciente na versão *desktop*

A Tela de Odontograma representa o início da etapa de exames clínicos do paciente e tem como objetivo registrar, de forma visual e interativa, a condição de cada dente. Nessa interface, todos os dentes são apresentados seguindo a numeração odontológica padrão, o que permite identificar de maneira clara o quadrante e a posição de cada dente na arcada dentária.

Cada dente é representado por um elemento *SVG* clicável. Ao selecionar um dente, o sistema abre um *modal* contendo as opções de condições clínicas que podem estar associadas a ele. Nesta interface, cada condição clínica é acompanhada por um *checkbox* clicável, permitindo que o usuário selecione múltiplas opções para o mesmo dente de forma rápida e intuitiva. Essas opções incluem estados como ausência, higidez, restaurações e tratamentos indicados, refletindo diretamente a legenda do sistema. O botão de “Aplicar” confirma e salva as opções selecionadas no momento, atualizando visualmente o odontograma, o de “Cancelar” fecha o *modal* cancelando as opções e o botão “Reverter ao Original” funciona como um recurso de segurança. Caso o usuário abra um dente que já possui condições salvas e faça novas alterações, este botão restaura o estado do dente exatamente como estava antes da abertura atual do *modal*, anulando as modificações recentes.

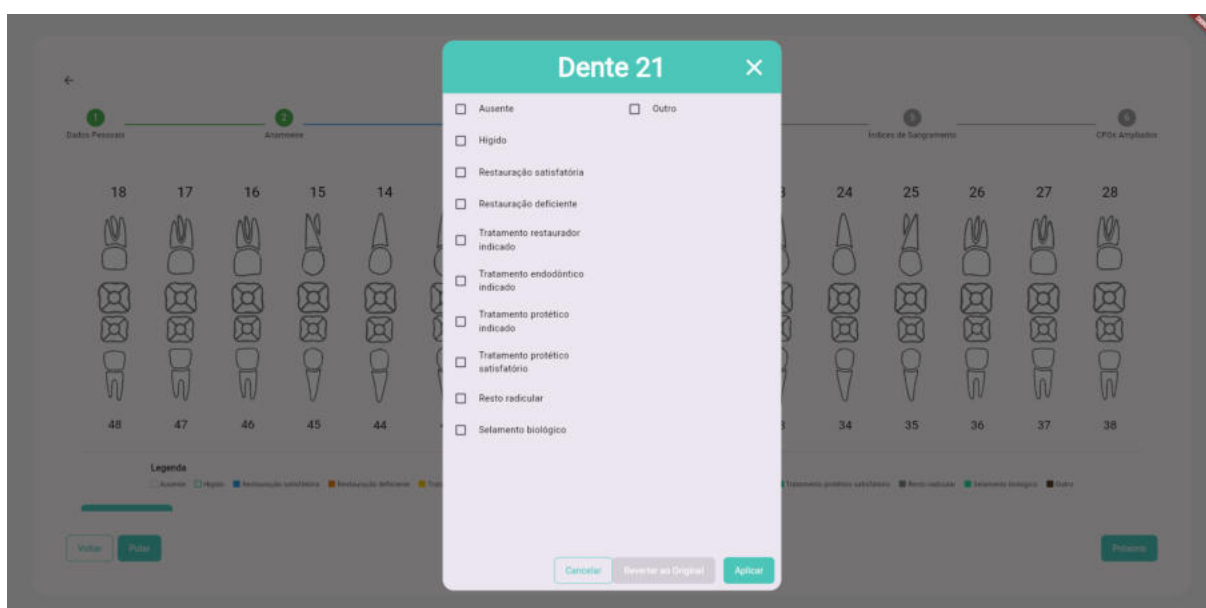


Figura 28 – *Modal* do dente 21 no odontograma

A legenda tem papel fundamental na interpretação do odontograma, pois associa cada condição clínica a uma cor específica. Quando uma ou mais opções são selecionadas para um dente, o *SVG* correspondente é preenchido ou contornado com a(s) cor(es) relacionada(s), permitindo uma visualização imediata e intuitiva do estado dentário do paciente. O sistema também permite que um mesmo dente possua mais de uma cor simultaneamente, representando a coexistência de diferentes condições clínicas.

Além das opções pré-definidas, a interface disponibiliza um botão de “Observações”, que abre um *modal* específico para o registro de informações adicionais. Esse recurso é utilizado para anotar particularidades clínicas ou observações relevantes que não se enquadram nas opções disponíveis no *modal* principal, garantindo maior flexibilidade e completude no registro das informações odontológicas, numa tentativa de simular a

liberdade que se tem quando a clínica utilizava o papel.

Assim como nas etapas anteriores do cadastro do paciente, o odontograma integra o fluxo guiado do sistema, permitindo que o usuário avance, retorne ou pule etapas conforme necessário.

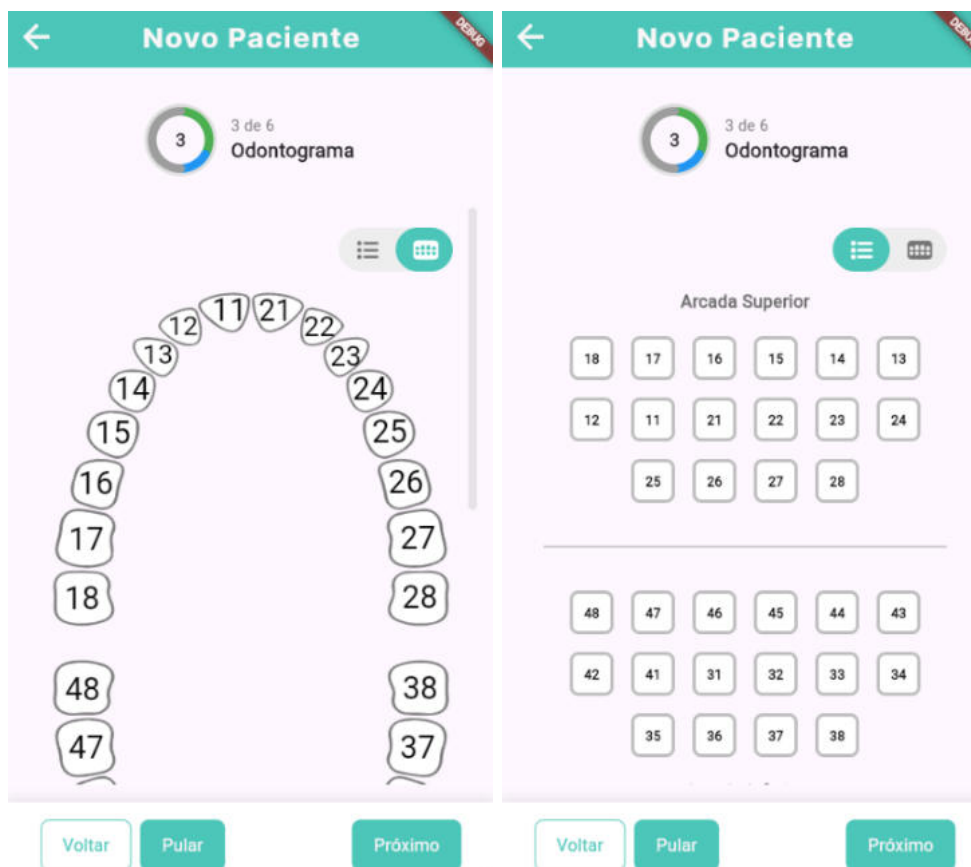


Figura 29 – Tela de odontograma de paciente na versão *mobile*

Na versão *mobile*, a interface do Odontograma foi otimizada para garantir a usabilidade em dispositivos de toque, apresentando duas formas distintas de interação que podem ser alternadas através de um seletor (*toggle*) posicionado no canto superior direito:

- Visualização Anatômica (Ícone de Arcada): Esta visualização mantém a representação gráfica e curvilínea da arcada dentária, similar à versão *desktop*. Os dentes são dispostos em formato de “O”, facilitando a identificação espacial rápida. Devido à limitação de largura da tela, o sistema utiliza o *scroll* vertical para permitir que o usuário navegue entre a arcada superior e inferior sem perder a clareza visual dos elementos *SVG*.
- Visualização em Grade/Lista (Ícone de Tabela): Para facilitar a seleção precisa em telas menores, o sistema oferece uma alternativa onde os dentes são organizados em uma grade. Os dentes são agrupados por quadrantes e alinhados horizontalmente,

transformando os elementos em botões maiores e mais fáceis de clicar com o polegar, reduzindo o erro de seleção acidental.

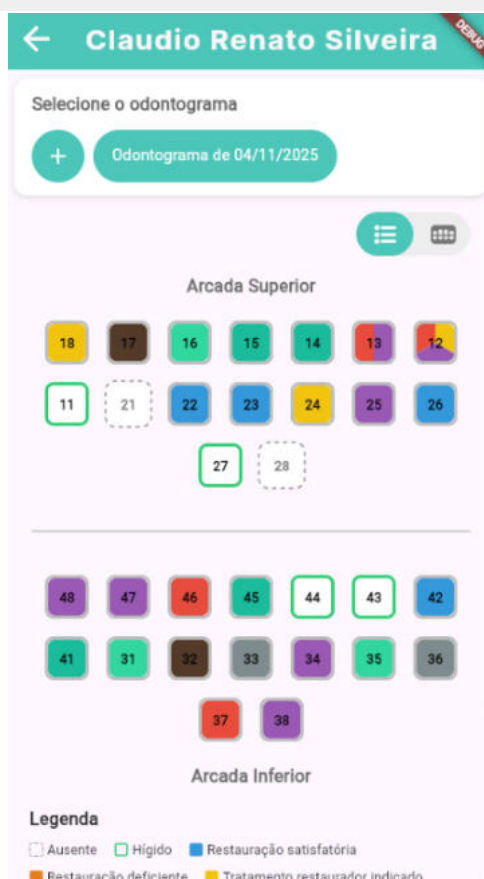
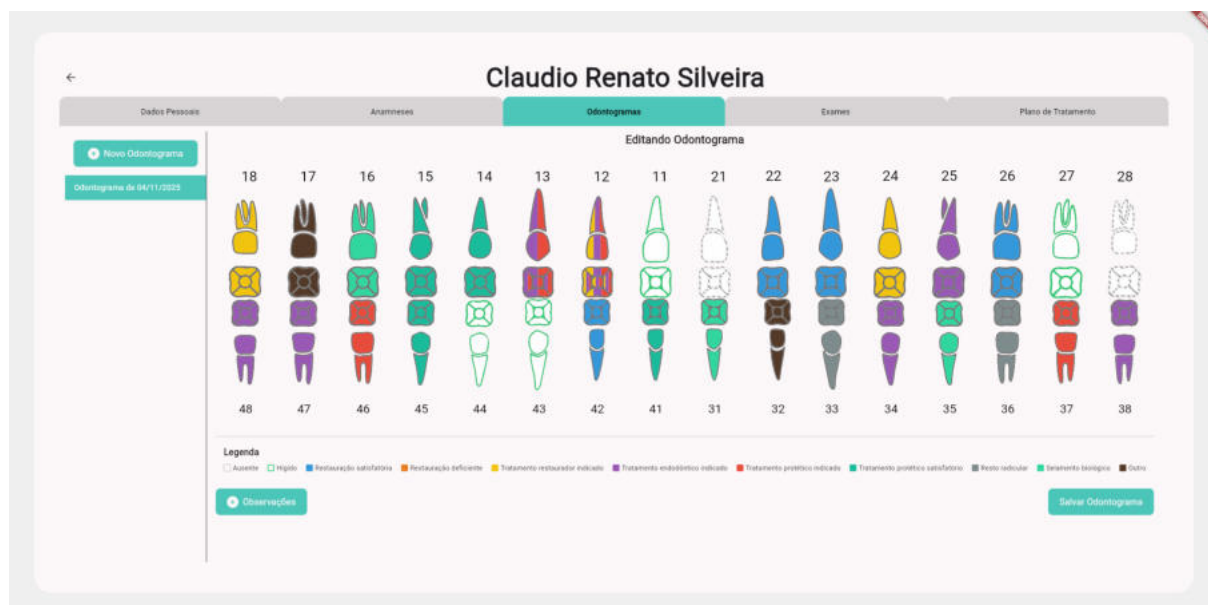


Figura 30 – Tela de lista de odontograma na versão *desktop* e *mobile*

Assim como ocorre com a Anamnese, o Odontograma pode ser consultado e editado a qualquer momento após a criação inicial do paciente. O acesso segue o mesmo padrão de

navegação nas versões *desktop* e *mobile*: na versão *desktop* (Figura 30), o usuário seleciona o paciente na listagem e navega até a aba “Odontograma”, enquanto na versão *mobile* o acesso é realizado por meio do *menu* do paciente, após a seleção do registro desejado.

O sistema também permite o registro de múltiplos odontogramas para um mesmo paciente, considerando que o estado dentário pode se alterar ao longo do tempo em decorrência de novos tratamentos ou procedimentos realizados em consultas posteriores. Dessa forma, cada odontograma representa um registro clínico associado a um determinado momento.

4.1.15 Tela de Paciente – Exame de Biofilme

Novo Paciente

1 Dados Pessoais 2 Anamnese 3 Odonograma 4 Índices de Biofilme Visível 5 Índices de Sangramento 6 CPOs Ampliados

18	17	16	15	14	13	12	11
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-

21	22	23	24	25	26	27	28
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-

48	47	46	45	44	43	42	41
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-

31	32	33	34	35	36	37	38
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-

Resultado % Fatores Determinantes - FR

Voltar Pular Próximo

Novo Paciente

4 de 6
Índices de Biofilme Visível

18	17	16	15	14	13	12	11
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-

21	22	23	24	25	26	27	28
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-

Voltar Pular Próximo

Figura 31 – Tela exames de biofilme de paciente na versão *desktop* e *mobile*

A Tela de Exame de Biofilme Visível é apresentado na forma de quatro tabelas distintas, cada uma representando um dos quadrantes da arcada dentária. Essa organização foi definida a partir de uma solicitação direta da *stakeholder*, com o objetivo de reproduzir

fielmente o modelo da ficha clínica, garantindo familiaridade aos profissionais e agilidade no preenchimento das informações.

Cada tabela é composta por colunas numeradas, em que cada coluna corresponde a um dente específico, identificado por sua numeração odontológica. As linhas da tabela representam as faces dentárias, identificadas pelas siglas D/M, V, M/D e P/L. A interpretação das faces D/M e M/D varia de acordo com o quadrante ao qual o dente pertence, respeitando a simetria da arcada dentária.

Nos quadrantes localizados no lado esquerdo do paciente (quadrantes 1 e 4), a primeira linha identificada como D/M representa a face distal do dente, enquanto a terceira linha, M/D, corresponde à face mesial. Por exemplo, na coluna referente ao dente 11, o campo D/M indica a face distal desse dente.

Já nos quadrantes localizados no lado direito do paciente (quadrantes 2 e 3), essa lógica é invertida. Nesses casos, a linha D/M passa a representar a face mesial, e a linha M/D corresponde à face distal. Assim, na coluna do dente 21, o campo D/M refere-se à face mesial do dente.

A face vestibular (V) mantém a mesma interpretação em todos os quadrantes. Já a face P ou L varia conforme a posição do dente na arcada: nos quadrantes superiores (1 e 2), corresponde à face palatina, voltada para o céu da boca, enquanto nos quadrantes inferiores (3 e 4) corresponde à face lingual, voltada para a língua.

Cada célula da tabela é interativa e permite o registro da presença ou ausência de biofilme. Ao selecionar uma célula, o valor é preenchido com "1", indicando a presença de biofilme visível naquela face dentária. Caso não haja biofilme, o valor registrado é "0". Quando o dente está ausente, as células correspondentes são exibidas com um padrão tracejado, impossibilitando a interação e indicando visualmente que aquele dente não deve ser considerado no exame.

Na versão *mobile*, a apresentação do exame de biofilme sofre uma adaptação visual com o objetivo de melhorar a legibilidade e a usabilidade em telas menores. Nesse contexto, as quatro tabelas correspondentes aos quadrantes da arcada dentária são exibidas de forma empilhada verticalmente, permitindo que o usuário visualize e preencha cada quadrante de maneira sequencial e organizada. Devido a essa reorganização, a nomenclatura das faces dentárias é apresentada de forma direta e explícita, eliminando a necessidade da notação composta D/M e M/D.

Além do preenchimento das faces dentárias, a tela do exame de biofilme dispõe de um campo de texto destinado à apresentação do resultado percentual do exame. Esse campo exibe o índice de biofilme do paciente, calculado a partir da relação entre as faces com presença de biofilme e o total de faces avaliadas. Esse valor percentual auxilia na avaliação da condição de higiene bucal do paciente e pode ser utilizado como indicador clínico para acompanhamento da evolução do tratamento ao longo do tempo.

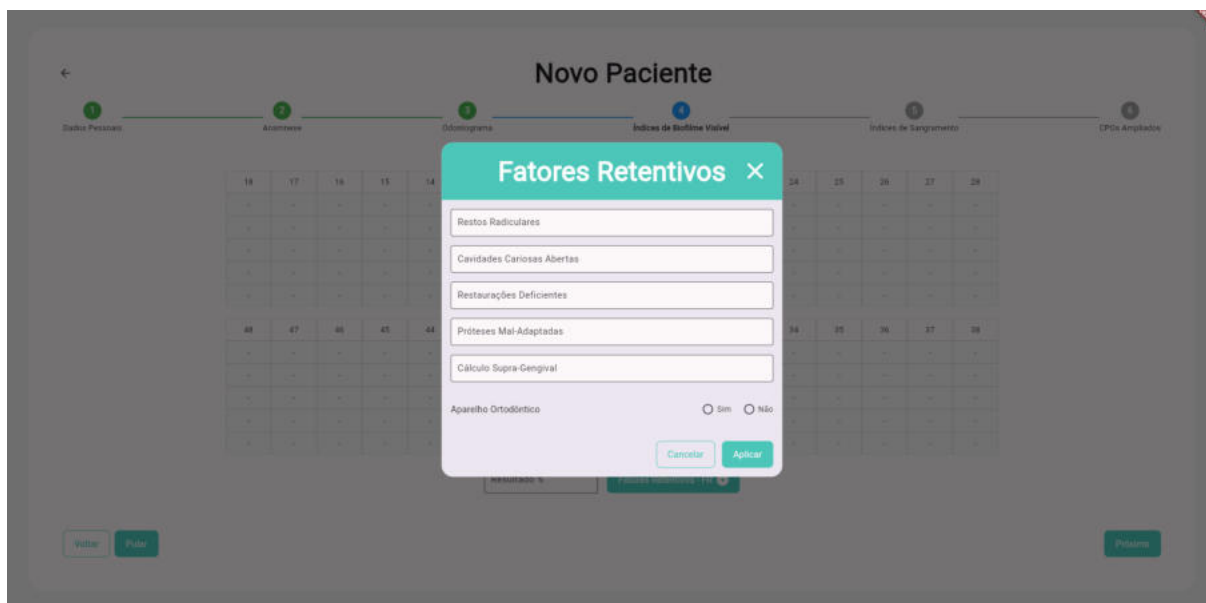


Figura 32 – *Modal* de Fatores Retentivos

Complementando o exame, a interface disponibiliza o botão “Fatores Retentivos”, que aciona a abertura de um *modal* específico para o registro de condições clínicas adicionais que favorecem o acúmulo de biofilme e dificultam a higienização como o uso de aparelhos ortodônticos, restos radiculares, cavidades cariosas abertas, restaurações deficientes, próteses mal-adaptadas e cálculo supra-gengival. Esses dados são coletados por meio de campos de texto livres, permitindo ao profissional descrever com maior precisão as condições observadas durante o atendimento.

A única exceção refere-se à presença de aparelhos ortodônticos, que é registrada por meio de um campo de seleção do tipo *radio button*, com as opções “Sim” e “Não”, seguindo o mesmo padrão visual e funcional adotado na tela de Anamnese.



Figura 33 – Tela da lista de exames biofilme na versão *desktop* e *mobile*

Após a criação inicial do paciente, os exames clínicos podem ser consultados e editados a qualquer momento por meio da seção “Exames”, disponível tanto na versão *desktop* quanto na versão *mobile* do sistema. Em ambas as plataformas, o acesso ocorre após a seleção do paciente na listagem: na versão *desktop* (Figura 33), o usuário navega até a aba “Exames”, enquanto na versão *mobile* o acesso é realizado por meio do *menu* do paciente, selecionando a opção correspondente.

Dentro da seção de exames, o sistema apresenta três botões principais que permitem alternar entre os diferentes tipos de avaliação clínica disponíveis: Exame de Biofilme, Exame de Sangramento e Exame CPOS.

O Exame de Biofilme segue a mesma lógica de versionamento temporal adotada em outras partes da ficha clínica, permitindo o cadastro de múltiplos exames para um mesmo paciente.

4.1.16 Tela de Paciente – Exame de Sangramento

Figura 34 – Tela exames de sangramento de paciente na versão *desktop* e *mobile*

A Tela de Exame de Sangramento segue o mesmo padrão visual e estrutural adotado na Tela de Exame de Biofilme, tanto no *desktop* quanto no *mobile*, atendendo à solicitação da *stakeholder* para manter fidelidade à ficha clínica. A principal diferença entre os dois exames está na estrutura da tabela apresentada.

No Exame de Sangramento, a tabela não contempla a face oclusal (O), sendo composta apenas pelas faces periodontais relevantes para a avaliação do sangramento gengival.

Assim como nos demais exames clínicos, o preenchimento da tabela é realizado de forma interativa. Ao selecionar uma célula correspondente a uma face dentária, o sistema permite indicar a presença ou ausência de sangramento, registrando a informação de forma visual. E também há a presença do campo de texto para colocar o resultado do índice de sangramento.

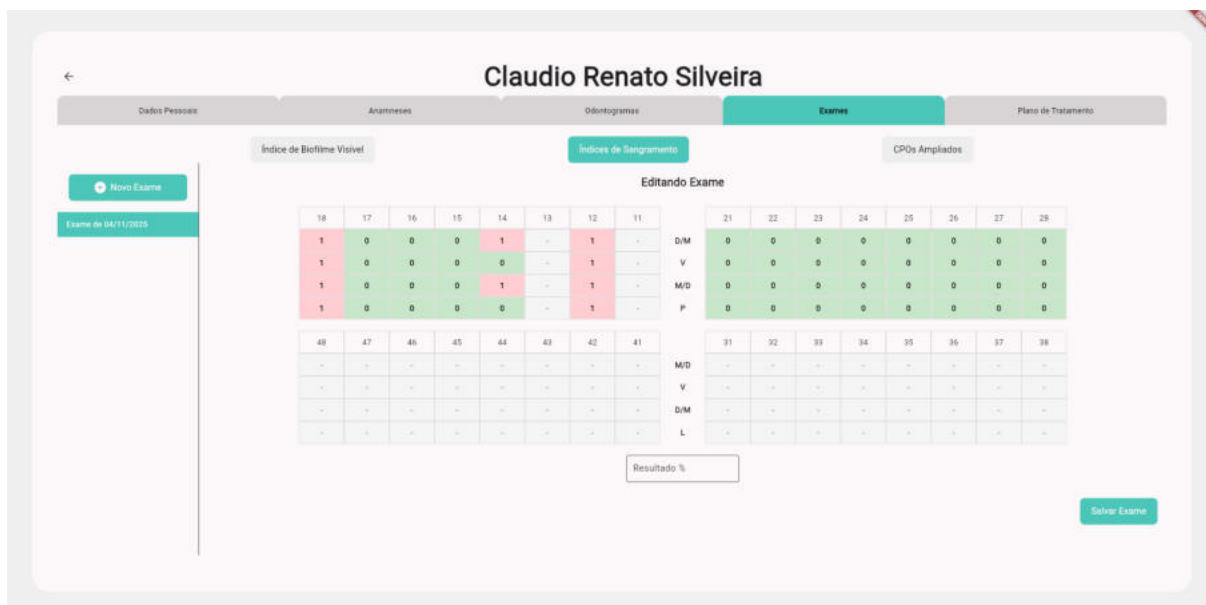


Figura 35 – Tela da lista de exames sangramento na versão *desktop* e *mobile*

Após a criação inicial do paciente, os exames clínicos podem ser consultados e editados a qualquer momento por meio da seção “Exames”, disponível tanto na versão *desktop* quanto na versão *mobile* do sistema. Em ambas as plataformas, o acesso ocorre após a seleção do paciente na listagem: na versão *desktop* (Figura 35), o usuário navega até a aba “Exames”, enquanto na versão *mobile* o acesso é realizado por meio do *menu* do paciente.

Dentro da seção de exames, o sistema disponibiliza três botões principais que permi-

tem alternar entre os diferentes tipos de avaliação clínica disponíveis: Exame de Biofilme, Exame de Sangramento e Exame CPOS. Essa organização possibilita uma navegação rápida entre os exames, mantendo a consistência da interface e reduzindo o tempo necessário para o registro das informações clínicas.

O Exame de Sangramento, assim como o Exame de Biofilme, adota a lógica de versionamento temporal, permitindo o cadastro de múltiplos exames para um mesmo paciente.

4.1.17 Tela de Paciente – Exame de CPOS

Novo Paciente

1 Dados Pessoais 2 Anamnese 3 Odonograma 4 Índices de Biofilme Visual 5 Índices de Sangramento 6 CPOs Ampliados

18	17	16	15	14	13	12	11	
-	-	-	-	-	-	-	-	D
-	-	-	-	-	-	-	-	V
-	-	-	-	-	-	-	-	M
-	-	-	-	-	-	-	-	P
-	-	-	-	-	-	-	-	O

21	22	23	24	25	26	27	28	
-	-	-	-	-	-	-	-	D
-	-	-	-	-	-	-	-	V
-	-	-	-	-	-	-	-	M
-	-	-	-	-	-	-	-	P
-	-	-	-	-	-	-	-	O

48	47	46	45	44	43	42	41	
-	-	-	-	-	-	-	-	M
-	-	-	-	-	-	-	-	V
-	-	-	-	-	-	-	-	D
-	-	-	-	-	-	-	-	L
-	-	-	-	-	-	-	-	O

31 32 33 34 35 36 37 38

-	-	-	-	-	-	-	-	D
-	-	-	-	-	-	-	-	V
-	-	-	-	-	-	-	-	M
-	-	-	-	-	-	-	-	P
-	-	-	-	-	-	-	-	O

Voltar Salvar

Novo Paciente

6 de 6 CPOs Ampliados

18	17	16	15	14	13	12	11	
-	-	-	-	-	-	-	-	D
-	-	-	-	-	-	-	-	V
-	-	-	-	-	-	-	-	M
-	-	-	-	-	-	-	-	P
-	-	-	-	-	-	-	-	O

21	22	23	24	25	26	27	28	
-	-	-	-	-	-	-	-	D
-	-	-	-	-	-	-	-	V
-	-	-	-	-	-	-	-	M
-	-	-	-	-	-	-	-	P
-	-	-	-	-	-	-	-	O

Voltar Salvar

Figura 36 – Tela exames cpos de paciente na versão *desktop* e *mobile*

O Exame de CPOS segue o mesmo padrão estrutural adotado nos Exames de Biofilme, tanto no *desktop* tanto no *mobile*, sendo apresentado por meio de tabelas organizadas em quadrantes, de acordo com a arcada dentária.

Diferentemente do Exame de Biofilme e do Exame de Sangramento, que utilizam marcações binárias para indicar presença ou ausência, o Exame de CPOS adota uma lógica semelhante à do Odontograma. Cada célula da tabela representa uma superfície específica de um dente e funciona como um elemento interativo.

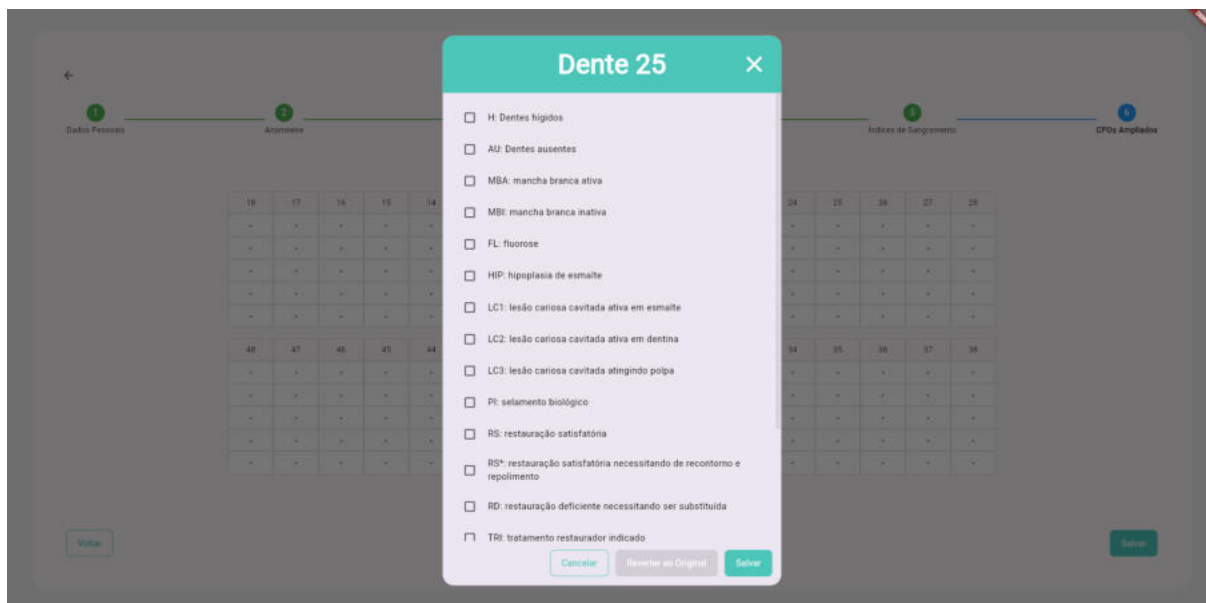


Figura 37 – *Modal* do dente 14 no exame cpos

Ao selecionar uma célula, o sistema abre um *modal* no qual o usuário pode escolher o estado clínico correspondente àquela superfície dentária, como hígido, cariado, restaurado ou perdido, de acordo com a condição observada durante o exame. Após a seleção, a informação é refletida visualmente na tabela por meio de uma identificação alfanumérica padronizada, permitindo uma leitura clara e rápida do exame. Por exemplo, superfícies hígidas são identificadas pela sigla H, enquanto uma lesão cáriosa cavitada ativa em esmalte é representada por LC1.

Nos casos em que múltiplas condições coexistem em uma mesma superfície, o sistema exibe um indicador numérico correspondente à quantidade de registros associados ao dente.



Figura 38 – Tela da lista de exames cpos na versão *desktop* e *mobile*

Após a criação inicial do paciente, os exames de CPOS podem ser consultados e editados a qualquer momento por meio da seção “Exames”, disponível tanto na versão *desktop* quanto na versão *mobile* do sistema. Em ambas as plataformas, o acesso ocorre após a seleção do paciente na listagem: na versão *desktop* (Figura 38), o usuário navega até a aba “Exames”, enquanto na versão *mobile* o acesso é realizado por meio do *menu* do paciente.

Dentro da seção de exames, o sistema disponibiliza três botões principais que permitem alternar entre os diferentes tipos de avaliação clínica disponíveis: Exame de Biofilme, Exame de Sangramento e Exame de CPOS.

O Exame de CPOS, assim como os demais exames clínicos do sistema, adota a lógica de versionamento temporal. Dessa forma, é possível registrar múltiplos exames de CPOS para um mesmo paciente ao longo do tempo, preservando o histórico clínico e permitindo o acompanhamento da evolução da condição dentária em diferentes consultas.

4.2 TRABALHOS RELACIONADOS

Nesta seção são apresentados sistemas e soluções existentes que atuam no contexto de gestão odontológica digital, com o objetivo de compreender como diferentes plataformas abordam problemas semelhantes aos tratados neste trabalho. A análise de trabalhos relacionados permite identificar funcionalidades já consolidadas no mercado, bem como lacunas e oportunidades que podem ser exploradas no desenvolvimento de novas soluções.

Para a realização dessa análise foram selecionadas duas aplicações amplamente utilizadas no contexto odontológico: o *Dental Office* e o *Dental Escola*. A escolha dessas plataformas ocorreu a partir de uma investigação exploratória de sistemas de gestão odontológica disponíveis no mercado brasileiro, considerando critérios como relevância no setor, disponibilidade de documentação pública sobre suas funcionalidades e aderência ao contexto clínico e educacional relacionado ao projeto SISCAR.

4.2.1 Dental Office

O Dental Office é uma plataforma comercial consolidada de gestão odontológica voltada principalmente para clínicas e consultórios particulares (Dental Office, 2026b). Seu foco está na automação de processos administrativos, otimização da rotina de atendimento e aprimoramento da experiência do paciente.

Entre suas principais funcionalidades, destacam-se:

- **Agenda online:** permite o controle completo de atendimentos, com identificação de *status* por cores, separação de agenda por dentista ou cadeira e gerenciamento de consultas, retornos, faltas e fila de espera. Além disso, possibilita que o próprio paciente realize o agendamento de forma totalmente digital, por meio de links disponibilizados em sites, redes sociais ou aplicativos de mensagens.
- **Integração com o WhatsApp:** o sistema oferece confirmação automática de consultas e envio de mensagens personalizadas, reduzindo significativamente o número de faltas e melhorando o relacionamento com o paciente.

- **Prontuário eletrônico e ficha de especialidades:** disponibiliza registro clínico digital integrado ao histórico do paciente, contemplando diferentes especialidades odontológicas e facilitando o acompanhamento contínuo do tratamento.
- **Gestão financeira:** inclui módulos de controle de fluxo de caixa, emissão de notas fiscais eletrônicas, gestão de receitas e despesas, cálculo de comissões e geração de relatórios gerenciais.
- **Marketing e relacionamento:** disponibiliza ferramentas voltadas à fidelização de pacientes, acompanhamento de leads e campanhas automatizadas para manutenção do relacionamento.
- **Recursos de inteligência artificial:** oferece suporte à análise diagnóstica e comandos por assistente de voz, visando aumentar a agilidade e a precisão no atendimento.

4.2.2 Dental Escola

O *Dental Escola*, versão acadêmica do *Dental Office*, é uma plataforma desenvolvida para instituições de ensino odontológico, com foco na rotina de clínicas-escola e no acompanhamento da evolução dos alunos (Dental Office, 2026a). Diferentemente de outras plataformas comerciais voltadas ao mercado privado, este módulo incorpora recursos voltados à supervisão e validação de procedimentos em ambiente educacional.

Entre seus principais recursos, destacam-se:

- **Acompanhamento da evolução e produtividade dos alunos:** permite o controle detalhado do desempenho discente, auxiliando professores e gestores no acompanhamento individual e coletivo das atividades clínicas.
- **Validação de procedimentos pelos professores:** os docentes podem validar atendimentos diretamente por dispositivos móveis, garantindo agilidade e rastreabilidade das supervisões.
- **Agenda odontológica digital:** oferece controle completo das consultas e atividades clínicas, assegurando pontualidade e organização.
- **Acompanhamento da saúde periodontal:** inclui o *Periograma*, ferramenta que permite identificar e monitorar doenças periodontais de forma gratuita.
- **Prontuário eletrônico integrado:** possibilita acesso rápido às informações dos pacientes, otimizando o tempo de atendimento e reduzindo a necessidade de registros manuais.

- **Segurança e conformidade legal:** garante proteção dos dados sensíveis e atendimento às exigências da Lei Geral de Proteção de Dados (LGPD), promovendo maior confiabilidade institucional.

4.2.3 Diferenças e semelhanças em relação ao SISCAR

Embora as soluções comerciais analisadas ofereçam funcionalidades avançadas de gestão clínica, há diferenças significativas quando comparadas ao contexto e aos objetivos do sistema SISCAR, voltado exclusivamente a uma clínica que presta atendimento gratuito e cumpre papel formativo na graduação em Odontologia.

- **Foco institucional/educacional vs. foco comercial:** plataformas como Dental Office concentram-se em necessidades de mercado (faturamento, emissão de notas, controle de receitas, marketing). O SISCAR, por outro lado, por ser uma clínica dentro de uma instituição de ensino que atende gratuitamente moradores, servidores e estudantes não exige módulos financeiros complexos.
- **Modelagem das fichas clínicas:** o SISCAR foi desenvolvido a partir da tradução direta das fichas físicas utilizadas pela própria clínica, garantindo aderência total à realidade do atendimento deles.
- **Controle de supervisão, acesso e registros acadêmicos:** em um ambiente de uma clínica atuando dentro de uma faculdade, é essencial que o sistema permita o controle rigoroso de acesso às informações, definindo claramente os níveis de permissão entre secretarias, alunos e professores. Além disso, o armazenamento estruturado dessas informações possibilita sua utilização em pesquisas futuras e trabalhos de conclusão de curso (TCCs) dentro da própria instituição, viabilizando análises clínicas e estatísticas de forma prática. Diferentemente do cenário atual, em que fichas em papel dificultam a coleta de dados e comprometem a integridade das informações, o SISCAR propicia uma base digital organizada e acessível, favorecendo tanto a gestão pedagógica quanto a produção científica, diferente do Dental Office e Escola.
- **Validação manual de procedimentos:** diferentemente do *Dental Escola*, que permite validação digital via dispositivos móveis, o SISCAR realiza atualmente essa etapa de forma manual, o que representa uma oportunidade de evolução futura.
- **Acompanhamento da evolução e produtividade dos alunos:** funcionalidade ainda não implementada no SISCAR. Esse tipo de recurso aplicado no Dental Escola permite o monitoramento detalhado do desempenho discente, possibilitando que professores acompanhem a evolução individual e coletiva dos alunos ao longo dos atendimentos clínicos. A adoção de um módulo desse tipo no futuro poderia

fortalecer o caráter pedagógico do sistema, integrando indicadores de produtividade, desempenho e evolução prática às demais informações clínicas registradas.

- **Processo de desenvolvimento participativo:** o SISCAR foi concebido a partir do *Design Participativo* e do uso de metodologias ágeis (*Scrum*), garantindo que sua modelagem conceitual e estrutura de dados fossem continuamente validadas junto aos *stakeholders* (professores). Esse processo assegura maior aderência à realidade da instituição e promove uma aprendizagem mútua entre desenvolvedores e usuários.

5 CONCLUSÃO

Este trabalho apresentou o desenvolvimento do sistema SISCAR, um prontuário eletrônico voltado para a realidade da Clínica de Atendimento Referenciado (CAR). O sistema foi concebido a partir de um levantamento detalhado de requisitos junto aos principais *stakeholders*, buscando compreender os fluxos de atendimento, as necessidades acadêmicas e as particularidades do ambiente clínico.

A proposta do trabalho surgiu da necessidade de auxiliar e minimizar as dificuldades enfrentadas pela clínica em decorrência do uso exclusivo de registros em papel. A gestão manual de fichas clínicas, prontuários, exames e agendamentos mostrou-se ineficiente diante do crescimento da clínica, resultando em lentidão nos processos, inconsistências de informações e dificuldades no acesso ao histórico dos pacientes. Nesse contexto, o SISCAR foi projetado como uma solução para centralizar, organizar e digitalizar essas informações, promovendo maior confiabilidade, rastreabilidade e eficiência operacional.

O desenvolvimento do sistema adotou uma abordagem centrada no usuário, fundamentada em práticas de *design* participativo, nas quais professores, alunos e funcionários da clínica contribuíram ativamente para a definição dos requisitos e validação das soluções propostas. Essa abordagem permitiu alinhar o sistema às necessidades reais da CAR, respeitando suas rotinas, limitações e objetivos acadêmicos.

A aplicação foi desenvolvida com foco nas funcionalidades essenciais, seguindo uma estratégia próxima ao conceito de *MVP* (*Minimum Viable Product*), priorizando módulos como cadastro de pacientes, prontuário eletrônico, exames odontológicos, odontograma, plano de tratamento e agendamento de consultas. Essas funcionalidades foram especificadas e validadas por meio de fluxos de usuário, modelagens conceituais e prototipagem de interfaces, garantindo uma base sólida para a implementação.

Do ponto de vista técnico, o sistema foi implementado utilizando tecnologias modernas e amplamente adotadas no mercado. O *backend* foi desenvolvido com *Express.js*, integrado a um banco de dados *PostgreSQL*, enquanto o *frontend* foi implementado com *Flutter*, possibilitando uma aplicação multiplataforma. A arquitetura foi complementada com o uso de *Docker* para padronização do ambiente de execução e com práticas de desenvolvimento orientado a testes (*TDD*), utilizando *Jest* e testes de integração, assegurando a confiabilidade das funcionalidades implementadas.

Os resultados obtidos demonstram que o sistema foi desenvolvido conforme o planejado, com seus principais módulos testados e validados. Os testes realizados contemplaram tanto cenários de uso esperado quanto situações de erro, permissões inválidas e dados inconsistentes, contribuindo para a robustez da aplicação. Dessa forma, acredita-se que o SISCAR atinge um patamar satisfatório em relação aos objetivos propostos, oferecendo uma solução viável para a modernização da gestão da Clínica de Atendimento Referenci-

ado.

O código-fonte da solução está disponível publicamente no repositório do projeto: <https://gitlab.com/siscar>.

5.1 LIMITAÇÕES

Apesar dos resultados alcançados, o desenvolvimento e a implantação do sistema SISCAR enfrentaram algumas limitações relevantes. Uma das principais dificuldades esteve relacionada à hospedagem da aplicação. Apesar de diversas tentativas, não foi possível encontrar uma solução de hospedagem gratuita que comportasse adequadamente um sistema dessa magnitude, considerando requisitos como banco de dados persistente, segurança e disponibilidade.

Também houve esforços para viabilizar a hospedagem do sistema nos servidores institucionais da Universidade Federal do Rio de Janeiro (UFRJ), por meio de contatos com o setor de Tecnologia da Informação e Comunicação (TIC). No entanto, tais tentativas não obtiveram retorno efetivo, o que impossibilitou a adoção dessa alternativa durante o período de desenvolvimento do trabalho.

Além disso, por se tratar de um sistema destinado a uma clínica, é necessária uma manutenção contínua, tanto técnica quanto operacional. Essa característica demanda uma equipe dedicada para monitoramento, correções e evoluções do sistema. Considerando o contexto acadêmico e o encerramento do ciclo de graduação dos autores, tornou-se inviável garantir a continuidade direta da implementação e manutenção após a conclusão do curso.

Foram consideradas alternativas como a vinculação do projeto a iniciativas de extensão universitária, a exemplo de projetos como o DevMob, ou a busca por apoio institucional junto à Inova UFRJ (Agência de Inovação da UFRJ), órgão responsável por iniciativas de inovação e empreendedorismo na universidade, que poderia contribuir com orientação, com o objetivo de dar continuidade ao desenvolvimento. Entretanto, não foi possível estabelecer essa integração dentro do período disponível.

Diante desse cenário, a alternativa complementar encontrada foi a transferência do projeto para outro aluno ou grupo que possa dar prosseguimento ao sistema como continuidade acadêmica.

5.2 TRABALHOS FUTUROS

Como trabalhos futuros, diversas melhorias e expansões podem ser incorporadas ao sistema SISCAR. Uma funcionalidade relevante a ser implementada é a visualização estruturada de *logs* do sistema, permitindo maior transparência, auditoria e rastreabilidade das ações realizadas pelos usuários.

Outra evolução importante seria a implementação de mecanismos de exportação de dados e geração de gráficos e relatórios. Esses recursos poderiam apoiar atividades de pes-

quisa e produção acadêmica dos alunos de Odontologia, possibilitando análises estatísticas a partir dos dados clínicos registrados no sistema.

Também se destaca como trabalho futuro a capacitação dos integrantes da clínica para o uso adequado do sistema, por meio de treinamentos e manuais, garantindo maior adesão e melhor aproveitamento das funcionalidades oferecidas. Por fim, a busca por uma solução de hospedagem mais estável e permanente permanece como um ponto fundamental para viabilizar o uso contínuo do sistema em ambiente real de produção.

REFERÊNCIAS

- AMSTEL, F. M. van. Políticas de participação no design de interação. 2009. Disponível em: https://www.usabilidoido.com.br/arquivos/politicas_participacao.pdf.
- BECK, K. et al. Manifesto for agile software development. 2001.
- BENOIT BUKIET FRÉDÉRIC, D. J.-C. B. Current state of dental informatics in the field of health information systems: a scoping review. 2022. Disponível em: <https://bmcoralhealth.biomedcentral.com/articles/10.1186/s12903-022-02163-9>.
- BJÖGVINSSON PELLE EHN, P.-A. H. E. Design things and design thinking: Contemporary participatory design challenges. 2012. Disponível em: <https://direct.mit.edu/desi/article/28/3/101/69069/Design-Things-and-Design-Thinking-Contemporary>.
- CHERRY, R. D. M. C. The importance of context in information system design: An assessment of participatory design. 1999. Disponível em: <https://link.springer.com/article/10.1007/s007660050017>.
- Dental Office. **Dental Escola: sistema para clínicas-escola de odontologia**. 2026. <https://www.dentaloffice.com.br/dental-escola/>. Acesso em: 27 fev. 2026.
- Dental Office. **Sistema de gestão para clínicas odontológicas**. 2026. <https://www.dentaloffice.com.br/>. Acesso em: 27 fev. 2026.
- DINGSOYRA, T. et al. A decade of agile methodologies: Towards explaining agile software development. **Journal of Systems and Software**, 2012. Disponível em: <https://doi.org/10.1016/j.jss.2012.02.033>.
- EWEBWORLD. **Why Flutter Is Reshaping App Development in 2025 — And What You Need to Know**. 2025. Acesso em: 29 dez. 2025. Disponível em: <https://medium.com/@ewebworld/why-flutter-is-reshaping-app-development-in-2025-and-what-you-need-to-know-fd0335e2f749>.
- FILHO, C. H. **O que são Design Tokens?** 2020. Acessado em: 15 mar. 2026. Disponível em: <https://brasil.uxdesign.cc/o-que-sao-design-tokens-cd408431727d>.
- GAMMA, E. et al. **Design Patterns: Elements of Reusable Object-Oriented Software**. Reading, MA: Addison-Wesley, 1994. ISBN 978-0-20163-361-0.
- GOOGLE.
- GOSSMAN, J. **Introduction to Model/View/ViewModel pattern for building WPF apps**. 2005. Acessado em: 15 mar. 2026. Disponível em: <https://learn.microsoft.com/en-us/archive/blogs/johngossman/introduction-to-modelviewviewmodel-pattern-for-building-wpf-apps>.
- HARTSON, R.; PYLA, P. S. **The UX Book: Agile UX Design for a Quality User Experience**. [S.l.]: Morgan Kaufmann, 2018.
- HEID JOSEPH E. CHASTEEN, A. W. F. D. W. The electronic oral health record. 2002. Disponível em: <https://www.thejcdp.com/doi/pdf/10.5005/jcdp-3-1-1>.

IBARRA, M. C. **Design como correspondência Antropologia e participação na cidade**. [S.l.]: Editora UFPE, 2021.

IETF. **Introduction to JSON Web Tokens**. 2026. <https://www.jwt.io/introduction#what-is-json-web-token>. Acesso em: 15 mar. 2026.

INC., D. **Docker Documentation**. 2026. <https://docs.docker.com>. Acesso em: 15 mar. 2026.

KRUG, S. **Don't Make Me Think, A Common Sense Approach to Web Usability**. [S.l.]: New Riders Publishing, 2000.

LAXAAR. **Core features of Express.js**. 2024. Acesso em: 3 mar. 2026. Disponível em: <https://medium.com/@laxaar/core-features-of-express-js-67673dae92dd>.

LEDLEY, R. **A history of medical informatics in the United States, 1950 to 1990**. Porto Alegre: American Medical Informatics Association, 2022.

MARTIN, R. C. **Clean Architecture: A Craftsman's Guide to Software Structure and Design**. 1st. ed. [S.l.]: Prentice Hall, 2017.

MORENO, A. M.; ALHAMMAD, M. Integrating user experience into agile. 2022. Disponível em: www.researchgate.net/publication/360187273_Integrating_User_Experience_into_Agile_-_An_Experience_Report_on_Lean_UX_and_Scrum.

MULLER, M. J. **The Human-Computer Interaction Handbook**. [S.l.]: CRC Press, 2007.

PR7. **Seleção para atendimento em saúde bucal**. 2022. Acesso em: 31 nov. 2024. Disponível em: <https://politicasestudantis.ufrj.br/index.php/593-selecao-para-atendimento-em-saude-bucal>.

SCHLEYER, H. S. T. Dental informatics, a cornerstone of dental practice. 2001. Disponível em: <https://sci-hub.se/https://doi.org/10.14219/jada.archive.2001.0237>.

SHVETS, A. **History of Design Patterns**. 2014. Acesso em: 11 nov. 2025. Disponível em: <https://refactoring.guru/design-patterns/history>.

SILVA, V. A. da. A lei que transforma a universidade, a universidade que transforma a sociedade. 2022. Disponível em: <https://conexao.ufrj.br/2022/04/a-lei-que-transforma-a-universidade-a-universidade-que-transforma-a-sociedade/>.

SILVEIRA, L. G. G. da et al. Avaliação do uso da informática pelo cd e acadêmico em odontologia. 2006. Disponível em: <https://revistargo.com.br/include/getdoc.php?id=345&article=147&mode=pdf>.

SONAWANE, S. **Exploring the Benefits of Flutter for Cross-Platform App Development**. 2025. Acesso em: 28 dez. 2025. Disponível em: https://flutternest.com/blog/flutter-for-cross-platform-app-development?utm_source=chatgpt.com.

WOOD, T. M.; KOMPARE, C. Participatory design methods for collaboration and communication. **code4lib**, 2017. Disponível em: <https://journal.code4lib.org/articles/12127>.

**ANEXO A – FICHA DO PACIENTE UTILIZADA NA CLÍNICA DE
ATENDIMENTO REFERENCIADO**

Questões Odontológicas	SIM	NÃO
1. Quantas vezes por dia escova os dentes?		
2. Usa fio dental? Com que frequência?		
3. Limpa a língua regularmente? Com o que?		
4. Faz uso de enxaguatórios bucais? Qual?		
5. Já extraiu algum dente? Por quê? Correu tudo bem?		
6. Suas gengivas sangram? Já fez algum tratamento periodontal?		
7. Já usou algum aparelho ortodôntico? Qual? Por quanto tempo?		
8. Você usa alguma prótese dentária? Há quanto tempo?		
9. Você costuma respirar pela boca? Sente sua boca seca?		
10. Você range ou aperta os dentes?		
11. Seu maxilar estala ou incomoda seus ouvidos? Sente dores?		
12. Você consome com frequência algum alimento muito escuro/corantes?		
13. Você costuma comer entre as refeições? Ingerir muito doce?		
14. Você está satisfeito com a cor, forma e alinhamento dos seus dentes?		

Observações:

Data: ___/___/___ Assinatura do professor: _____

Data: ___/___/___ Assinatura do professor: _____

Eu, _____ declaro serem verdadeiras as informações por mim prestadas contidas neste questionário. Declaro ainda, não ter omitido nenhuma informação sobre a história da minha saúde.

Rio de Janeiro, ____ de _____ de _____.

Assinatura do cliente ou responsável legal

Assinatura/carimbo do professor responsável

Assinatura do aluno



UFRJ - Centro de Ciências da Saúde. Faculdade de Odontologia
Departamento de Odontologia Social e Preventiva. Estágio Supervisionado - CAR
 Nome do paciente: _____ Telefone: _____
 Idade: _____ Endereço: _____
 email: _____
 Raça/cor: () preto () pardo () branco () outros

Questões Médicas	SIM	NÃO
1. Você esta sob cuidados médicos? Por quê?		
2. Você está tomando algum medicamento? Qual?		
3. Você tem alergia a alguma substância?		
4. Você já se submeteu a alguma cirurgia?		
5. Você tem problemas cardíacos?		
6. Você já teve algum infarto, derrame, AVC ?		
7. Você é hipertenso? Como é sua pressão arterial?		
8. Você já teve convulsões ou desmaios?		
9. Você tem asma, bronquite ou rinite alérgica?		
10. Você já teve hepatite ou tem problemas hepáticos?		
11. Você tem algum problema renal?		
12. Você tem diabetes?		
13. Você tem alguma disfunção da tireóide?		
14. Você tem ou já teve câncer?		
15. Você tem algum problema sanguíneo?		
16. Já fez transfusão sanguínea?		
17. Você fuma cigarros? Quantos?		
18. Consome bebidas alcoólicas? Qual frequência?		
19. Faz uso de drogas (entorpecentes, estimulantes, inibidores de apetite, etc.)?		
20. Você se considera uma pessoa tensa ou nervosa?		
21. Você pratica esportes radicais, de impacto ou lutas?		
22. Você está grávida?		
23. Existe mais alguma consideração, queixa ou doença que eu não tenha perguntado e você acha importante relatar?		



UFRJ – Centro de Ciências da Saúde
Faculdade de Odontologia
Departamento de Odontologia Social e Preventiva
Estágio Supervisionado - CAR

PROTOCOLO DE ATENÇÃO BÁSICA EM SAÚDE - CONTROLE DO PROCESSO SAÚDE-DOENÇA CÁRIE/PERIODONTAL (WEYNE ET AL)

Cliente: _____
 Matrícula: _____ Aluno: _____

EXAMES INICIAIS
Índice de Biofilme Visível – IBV

18	17	16	15	14	13	12	11		21	22	23	24	25	26	27	28
								D/M								
								V								
								M/D								
								P								
					---	---	---	O	---	---	---					
48	47	46	45	44	43	42	41		31	32	33	34	35	36	37	38
								M/D								
								V								
								D/M								
								L								
					---	---	---	O	---	---	---					

Faces com biofilme =1; Faces sem biofilme visível=0; Dentes ausentes=X
Resultado em %= _____ Data: ___/___/___ . Ass prof: _____ .
Ass aluno: _____ .

Índice de Sangramento Gingival – ISG

18	17	16	15	14	13	12	11		21	22	23	24	25	26	27	28
								D/M								
								V								
								M/D								
								P								
48	47	46	45	44	43	42	41		31	32	33	34	35	36	37	38
								M/D								
								V								
								D/M								
								L								

Áreas com sangramento gengival=1; Áreas sem sangramento gengival=0; Dentes ausentes=X
Resultado em %= _____ Data: ___/___/___ . Ass prof: _____ .
Ass aluno: _____ .

Como calcular os índices:
Número total de superfícies presentes=100%
Número total de superfícies com biofilme/sangramento=X
Resultado obtido através de uma regra de 3.

Fatores Retentivos de Biofilme - FRB

FRP	Elementos/faces
Restos radiculares	
Cavidades cariosas abertas	
Restaurações deficientes	
Próteses mal-adaptadas	
Cálculo supra-gengival	
Aparelho ortodôntico	() sim () não

Data: ___/___/___ Ass prof: _____ Ass aluno: _____

SOMENTE APÓS PROFILAXIA E USO DE FIO DENTAL.....

CPOS ampliado

18	17	16	15	14	13	12	11		21	22	23	24	25	26	27	28
								D								
								V								
								M								
								P								
								O								
48	47	46	45	44	43	42	41		31	32	33	34	35	36	37	38
								M								
								V								
								D								
								L								
								O								

Data: ___/___/___ Ass prof: _____ Ass aluno: _____

LEGENDA:

- H- Dentes hígidos
- AU- Dentes ausentes
- MBA- mancha branca ativa
- MBI- mancha branca inativa
- FL- fluorose
- HIP-hipoplasia de esmalte
- LC1- lesão cariosa cavitada ativa em esmalte
- LC2- lesão cariosa cavitada ativa em dentina
- LC3- lesão cariosa cavitada atingindo polpa
- PI- selamento biológico
- RS- restauração satisfatória
- RS*- restauração satisfatória necessitando de recontorno e repolimento
- RD- restauração deficiente necessitando ser substituída
- TRI- tratamento restaurador indicado
- TEI- tratamento endodôntico indicado
- TPI- tratamento protético indicado
- TPS- tratamento protético satisfatório
- RR- resto radicular

Cliente apresenta atividade de doença cárie?

(.) sim () não

Cliente apresenta gengivite?

<input type="checkbox"/> sim <input type="checkbox"/> não	Fases de preparação, em meio local / preparo da base	Data - assinatura
	Radiografias solicitadas:	
	<input type="checkbox"/> panorâmica <input type="checkbox"/> periapical <input type="checkbox"/> bite-wing	
	Tentar aproximação	
	Aplicação do fio	
	<input type="checkbox"/> Terapêutica <input type="checkbox"/> reposição	
	Estimativa do número de aplicações	
	Remoção dos fatores retentivos de biofilme através de:	
	<input type="checkbox"/> Escovação supra-gengival	
	Fechamento provisório das cavidades	
	Tratamentos endodônticos	
	Curativos	
	Reconstruções provisórias, repositivas	
	Avaliação da estabilidade e fixação em função da	
	avaliação	
	Avaliação da atividade de doença?	
	<input type="checkbox"/> sim <input type="checkbox"/> não	

PLANO DE TRATAMENTO – FASE DE REEQUILÍBRIO DO MEIO

Dentes/faces	Fase de reequilíbrio do meio bucal / Preparo da boca	Data - assinatura
	Exposição do caso ao cliente	
	Educação em saúde/Atividades educativas Temas abordados:	
	Aplicação de flúor () Terapêutico () Reposição Estimativa do número de aplicações: Remoção dos fatores retentivos de biofilme através de:	
	Raspagem supra-gengival	
	Fechamento provisório das cavidades	
	Tratamentos endodônticos Exodontias	
	Recontornos, repolimentos, reparos	
	Selantes de cicatrículas e fissuras em lesões já cavitadas	
	Houve controle da atividade de doença? () sim () não	

PLANO DE TRATAMENTO – FASE REABILITADORA

ODONTOGRAMA INICIAL

18		38	
17		37	
16		36	
15		35	
14		34	
13		33	
12		32	
11		31	
21		41	
22		42	
23		43	
24		44	
25		45	
26		46	
27		47	
28		48	

- H - HÍGIDO
- AU - AUSENTE
- LC - LESÃO CARIOSA
- RS - RESTAURAÇÃO SATISFATÓRIA
- RD - RESTAURAÇÃO DEFICIENTE
- TRI- TRATAMENTO RESTAURADOR INDICADO
- TEI- TRATAMENTO ENDODÔNTICO INDICADO
- TPI- TRATAMENTO PROTÉTICO INDICADO
- TPS- TRATAMENTO PROTÉTICO SATISFATÓRIO
- RR- RESTO RADICULAR
- PI- SELAMENTO BIOLÓGICO

REAVALIAÇÕES / REVISÕES

Cliente: _____
 Matrícula: _____
 Aluno: _____

Índice de Biofilme Visível - IBV

18	17	16	15	14	13	12	11		21	22	23	24	25	26	27	28
								D/M								
								V								
								M/D								
								P								
					---	---	---	O	---	---	---					
48	47	46	45	44	43	42	41		31	32	33	34	35	36	37	38
								M/D								
								V								
								D/M								
								L								
					---	---	---	O	---	---	---					

Faces com biofilme visível=1

Faces sem biofilme visível=0

Dentes ausentes=X

Resultado em % = _____ **Data:** ___/___/____. **Ass prof:** _____
Ass aluno: _____

Índice de Sangramento Gengival - ISG

18	17	16	15	14	13	12	11		21	22	23	24	25	26	27	28
								D/M								
								V								
								M/D								
								P								
48	47	46	45	44	43	42	41		31	32	33	34	35	36	37	38
								M/D								
								V								
								D/M								
								L								

Áreas com sangramento gengival=1

Áreas sem sangramento gengival=0

Dentes ausentes=X

Resultado em % = _____ **Data:** ___/___/____. **Ass prof:** _____
Ass aluno: _____

Cliente apresenta atividade de doença cárie?

() sim () não

Está havendo inativação das MBAs? Em caso positivo, indique o (s) dente (s) e face(s)

() sim () não

Cliente apresenta gengivite?

() sim () não

Houve consultas de reforço?() sim

() não



**UFRJ - Centro de Ciências da Saúde - Faculdade de Odontologia
Departamento de Odontologia Social e Preventiva - CAR**

CLIENTE: _____ SIAPE/DRE: _____

Termo de Esclarecimento

A Clínica de Atendimento Referenciado (CAR) da Faculdade de Odontologia da UFRJ funciona de segunda à sexta feira, de 9:00 às 12:00 hs e de 13:00 às 16:00 hs e em cada semestre do ano segue o calendário acadêmico da UFRJ, já que é parte do Estágio Supervisionado do Curso de Odontologia.

O atendimento preferencial é para funcionários da UFRJ e para alunos da UFRJ, com prioridade aos provenientes das políticas afirmativas.

A Clínica realiza procedimentos relacionados à atenção primária em Saúde Bucal e sua filosofia de atuação é trabalhar promovendo saúde e estimulando os auto cuidados, que devem ser realizados diariamente, como controle da dieta, escovação dentária, utilização de fio dental e bochechos com soluções fluoretadas. No atendimento clínico oferecemos os seguintes procedimentos:

Exame clínico e anamnese; Educação em saúde; Indicadores de biofilme (placa bacteriana) e sangramento gengival; Profilaxia; Urgência; Recontornos e repolimentos de restaurações; Raspagem supragengival; Exodontia; Acesso endodôntico; Fechamento provisório de cavidade; Tratamento Restaurador Atraumático; Aplicações Tópicas de Fluoreto; Restauração direta em resina ou em amálgama; Manutenção Periódica do tratamento.

Deve-se salientar que procedimentos de média e alta complexidade devem ser procurados nas demais Clínicas da Faculdade de Odontologia ou fora da Instituição. Também é importante ressaltar que os supervisores e funcionários da CAR não têm gerência ou responsabilidade sobre a continuidade dos tratamentos quando não se enquadrem no perfil da Clínica.

Caso o paciente falte duas vezes consecutivas, sem justificativa, perderá seu direito ao atendimento.

Como a clínica é de pequeno porte, possuindo apenas três consultórios, nem sempre será possível atender o paciente semanalmente, o que pode causar certa demora na finalização do tratamento.

A Clínica adota os procedimentos de biossegurança recomendados visando minimizar os riscos de contaminação por doenças no atendimento odontológico, no entanto, os mesmos ainda existem.

Termo de Consentimento Livre e Esclarecido

Por este instrumento particular declaro para efeitos éticos e legais, que eu

RG: _____ SIAPE/DRE: _____, concordo

em absoluta consciência com a realização do tratamento na CAR, observando que fui amplamente informado (a) e esclarecido (a) sobre as normas de funcionamento da mesma. Por estar de pleno acordo com o teor do presente termo, segue por mim datado e assinado.

Rio de Janeiro, _____ de _____ de _____.

(assinatura do paciente)

(assinatura do supervisor)



UFRJ - Centro de Ciências da Saúde
Faculdade de Odontologia
Departamento de Odontologia Social e Preventiva
CAR
Diário de dieta

Anote tudo que ingeriu durante três dias, incluindo água e medicamentos. Se necessário, use folhas extras.

Café da manhã: _____

Café da manhã: _____

Café da manhã: _____

Entre o café e o almoço: _____

Entre o café e o almoço: _____

Entre o café e o almoço: _____

Almoço: _____

Almoço: _____

Almoço: _____

Entre o almoço e o lanche: _____

Entre o almoço e o lanche: _____

Entre o almoço e o lanche: _____

Lanche: _____

Lanche: _____

Lanche: _____

Entre o lanche e o jantar: _____

Entre o lanche e o jantar: _____

Entre o lanche e o jantar: _____

Jantar: _____

Jantar: _____

Jantar: _____

Após o jantar: _____

Após o jantar: _____

Após o jantar: _____

ANEXO B – CONTRATO ENTRE A CAR E A SUPEREST (PR-7)



**UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
GABINETE DO REITOR
SUPERINTENDÊNCIA GERAL DE POLÍTICAS ESTUDANTIS
DIVISÃO DE SAÚDE DO ESTUDANTE**

Termo de Cooperação de Assistência que, entre si, celebram a Superintendência Geral de Políticas Estudantis (SuperEst), através da Divisão de Saúde do Estudante (DISAE) e a Faculdade de Odontologia (FO), através do Departamento de Odontologia Social e Preventiva - Clínica de Atendimento Referenciado

Por este TERMO DE COOPERAÇÃO DE ASSISTÊNCIA, de um lado, a Superintendência Geral de Políticas Estudantis - SuperEst inscrita no CNPJ sob o n.º 33.663.683/0001-16, através da Divisão de Saúde do Estudante (DISAE), localizada na Avenida Pedro Calmon, 550 - Cidade Universitária - Ilha do Fundão, no Município do Rio de Janeiro - RJ, representado neste ato pelo Coordenador Superintendente Prof. Dr. Antônio José Barbosa de Oliveira - SIAPE N.º 2124782 e, do outro lado, a Faculdade de Odontologia inscrita no CNPJ sob o n.º 33.663.683/0016-00, através do Departamento de Odontologia Social e Preventiva - Clínica de Atendimento Referenciado, localizada na Avenida Carlos Chagas Filho, 373 - Prédio do CCS, Bloco K, segundo andar, sala 56 - Cidade Universitária - Ilha do Fundão, no Município do Rio de Janeiro - RJ, neste ato representado pelo Coordenador Diretor Prof. Dr. Ednilson Porangaba Costa - SIAPE N.º 0368738, resolvem firmar o presente Termo de Cooperação de Assistência, de acordo com as seguintes cláusulas e condições que se seguem:

CLÁUSULA PRIMEIRA - OBJETO

O presente Termo de Cooperação de Assistência visa a estabelecer colaboração mútua entre a Superintendência Geral de Políticas Estudantis (SuperEst) e a Faculdade de Odontologia (FO), através da Divisão de Saúde do Estudante (DISAE) e o Departamento de Odontologia Social e Preventiva - Clínica de Atendimento Referenciado, respectivamente, de forma a contribuir na constituição de uma rede de cuidados à saúde no campo da Odontologia para promoção, prevenção, tratamento e reabilitação do corpo discente da UFRJ inserido nos programas de assistência estudantil da SuperEst abrangendo ações de promoção, prevenção, tratamento e reabilitação em saúde bucal.

Sem prejuízo de outras formas, a colaboração a que se refere o parágrafo anterior estará voltada, essencialmente, para a assistência aos estudantes inseridos nos programas de assistência estudantil da SuperEst estimados a priori, em 420 (quatrocentos e vinte) alunos até o mês de Dezembro de 2013, abrangendo ações de promoção, prevenção, tratamento e reabilitação em saúde bucal.

CLÁUSULA SEGUNDA - DAS OBRIGAÇÕES

Compete à Superintendência Geral de Políticas Estudantis (SuperEst), através da Divisão de Saúde do Estudante (DISAE), para atender aos fins deste Termo de Cooperação de Assistência, dentre outras atividades:

§ 1º Oferecer um espaço na página da Superest para divulgação e informação com relação a importância da saúde bucal para os alunos da Residência Estudantil da UFRJ;

§ 2º Oferecer 10 (dez) bolsas administrativas no valor de R\$ 400,00 (quatrocentos reais) com carga horária de 20 horas semanais no intuito de garantir pessoal para a assistência a aproximadamente 420 (quatrocentos e vinte) estudantes objetos desse Termo abrangendo ações de promoção, prevenção, tratamento e reabilitação em saúde bucal;

§ 3º Dar apoio material à Faculdade de Odontologia para que esta possa desenvolver atividades de promoção e educação em saúde (palestras, seminários, distribuição de materiais didáticos, cursos e campanhas educativas), tratamento e reabilitação em saúde bucal;

§ 4º Elaborar o Boletim Anual de Produção e Estatística (BAPE) referente aos procedimentos realizados pela Faculdade de Odontologia aos estudantes com problemas de saúde bucal encaminhados pela DISAE e relatórios semestrais do quantitativo de alunos atendidos; da avaliação do atendimento e seu impacto no período de integralização do estudante no Curso de Graduação.

Compete à Faculdade de Odontologia, para atender aos fins deste Acordo de Cooperação de Assistência, dentre outras atividades: .

§ 5º Promover e estimular, junto ao corpo discente da UFRJ, ações de promoção e educação em saúde (palestras, seminários, distribuição de materiais didáticos, cursos e campanhas educativas), que mobilizem e orientem quanto aos problemas de saúde bucal, reduzindo as chances e

condições que favoreçam o desenvolvimento de problemas relacionados à mesma;

§ 6º Detectar o problema de saúde bucal e motivar o estudante a aderir ao tratamento, ou ainda, a melhorar seu nível de informação sobre os riscos associados aos problemas de saúde bucal por meio de um aumento de seu senso de risco e de autocuidado. As metas são estabelecidas para cada estudante, individualmente, a partir da clara identificação do seu padrão atual de saúde bucal;

§ 7º Promover ações de promoção, prevenção, tratamento e reabilitação dos estudantes encaminhados pela DISAE/SuperEst;

§ 8º Disponibilizar espaço físico e profissional da área de Odontologia para a realização do atendimento e da assistência preconizados neste Termo;

§ 9º Garantir, em absoluto sigilo, a posse e a guarda dos documentos referentes aos pacientes encaminhados pela DISAE/SuperEst, atendendo aos preceitos e princípios éticos. A disponibilização dos documentos somente poderá ser realizada quando solicitado pelos responsáveis por este Termo e com o consentimento do usuário;

CLÁUSULA TERCEIRA - DA ADMINISTRAÇÃO

Os entendimentos necessários ao desenvolvimento das disposições deste Termo de Cooperação de Assistência serão mantidos pelos Coordenadores

das partes. Estes representantes serão competentes para o contato entre as duas Unidades.

CLÁUSULA QUARTA - DOS RECURSOS

As despesas necessárias à consecução do Objeto deste Termo serão assumidas pelos partícipes, dentro dos limites de suas respectivas atribuições.

CLÁUSULA QUINTA - DA VIGÊNCIA

O presente Termo de Cooperação de Assistência vigorará pelo prazo indeterminado, podendo ser rescindido ou alterado de comum acordo entre as partes, ou unilateralmente, por quaisquer delas na dependência de situações especiais ou extraordinárias, mediante Termo Aditivo, com antecedência mínima de 60 (sessenta) dias.

CLÁUSULA SEXTA - DA DENÚNCIA E RESCISÃO

O presente Termo de Cooperação de Assistência poderá ser denunciado por iniciativa de qualquer dos partícipes, mediante troca de avisos, com antecedência mínima de 60 (sessenta) dias ou rescindido por descumprimento de qualquer de suas cláusulas, assumindo cada partícipe os respectivos ônus decorrentes das obrigações acordadas. A eventual rescisão deste Termo de Cooperação de Assistência não prejudicará os projetos ou serviços já em andamento.

CLÁUSULA SETIMA - DAS DISPOSIÇÕES FINAIS

O presente Termo de Cooperação de Assistência só poderá ter alterado, em quaisquer de suas cláusulas, inclusive Objeto e Prazo, mediante Termo Aditivo firmado entre as partes. E, por estarem justas e acordadas, assinam as partes o presente Termo em 03 (três) vias, de igual teor e forma, para um só efeito, perante as testemunhas abaixo firmadas.

Rio de Janeiro, de

Superintendência Geral de Políticas Estudantis

Prof. Dr. Antônio José Barbosa de Oliveira
Superintendente - SuperEst

Prof.^a Dr.^a Marilurde Donato
Diretora da Divisão de Saúde do Estudante/SuperEst

Faculdade de Odontologia-FO

Prof. Dr. Ednilson Porangaba Costa
Diretor da Faculdade de Odontologia

Prof.^a Dr.^a Andreia Cristina Breda de Souza
Chefe do Departamento de Odontologia Social e Preventiva

Testemunhas:

_____ RG: _____

_____ RG: _____