

UNIVERSIDADE DE MARÍLIA

**AURAVET: PLATAFORMA WEB FULL STACK PARA
GESTÃO VETERINÁRIA: ARQUITETURA,
DESENVOLVIMENTO E DOCUMENTAÇÃO TÉCNICA**

ALUNO: Caio Murillo de Oliveira

CURSO: Análise e Desenvolvimento de Sistemas

DISCIPLINA: Projeto Integrador

PROFESSOR: Euandro Pinheiro de Oliveira

Sumário

1.	Introdução	5
1.1.	Contextualização	6
1.2.	Objetivo do Sistema.....	6
1.3.	Escopo do Projeto.....	7
1.4.	Público-Alvo	8
1.5.	Metodologia de desenvolvimento	9
1.6.	Cronograma de desenvolvimento.....	10
2.	Desenvolvimento.....	12
2.1.	Arquitetura do Sistema.....	13
2.1.1.	Estrutura de Diretórios	14
2.1.2.	Arquitetura em camadas	14
2.1.3.	Diagramas de Arquitetura.....	16
2.2.	Requisitos do sistema	18
2.2.1.	Requisitos funcionais	19
2.2.2.	Requisitos não funcionais.....	20
2.3.	Modelagem do sistema	21
2.3.1.	Casos de uso.....	21
2.3.2.	Diagrama de classe conceitual	34
2.3.3.	Diagramas de atividades.....	37
2.3.4.	Diagramas de sequência.....	48
2.4.	Tecnologias utilizadas.....	54
2.4.1.	Frontend	55
2.4.2.	Backend	55
2.4.3.	Banco de Dados.....	56
2.4.4.	Infraestrutura.....	56
2.4.5.	Justificativa das Tecnologias.....	56

2.5.	Branding e diretrizes visuais	57
2.5.1.	Paleta de cores	57
2.5.2.	Tipografia.....	58
2.5.3.	Logo.....	58
2.5.4.	Manual de UI.....	59
2.5.5.	Tom de voz	60
2.6.	Segurança do sistema.....	62
2.6.1.	Autenticação	62
2.6.2.	Autorização.....	64
2.6.3.	Proteções e Boas Práticas.....	65
2.7.	Testes automatizados	65
2.8.	Padrões de código e organização.....	68
2.8.1.	Convenções.....	69
2.8.2.	Linting e Formatação	70
2.8.3.	Hooks de desenvolvimento	71
2.9.	Guia de Deploy.....	71
2.9.1.	Execução Local.....	72
2.9.2.	Execução com Docker	73
2.10.	Referências de API	74
2.10.1.	Tabela Geral de Endpoints	75
2.10.2.	Fluxo de autorização no Swagger	76
2.10.3.	Exemplos de Chamadas	77
2.10.4.	Modelos de resposta e códigos de erro.....	82
2.11.	Telas.....	83
2.11.1.	Tela de Login.....	83
2.11.2.	Tela inicial (Dashboard)	84
2.11.3.	Tutores	85

2.11.4.	Pets.....	89
2.11.5.	Agendamentos.....	93
2.11.6.	Agenda Inteligente.....	104
2.11.7.	Serviços.....	107
2.11.8.	Produtos	111
2.11.9.	Atendimento.....	115
2.11.10.	Condições de pagamento.....	121
2.11.11.	Caixa.....	123
2.11.12.	Financeiro.....	130
2.11.13.	Usuários.....	135
2.11.14.	Funções	139
3.	Conclusão.....	143

1. Introdução

A Auravet nasce como uma clínica veterinária digital-first que combina ciência, tecnologia e acolhimento para oferecer um cuidado moderno e humanizado aos animais. Em um cenário em que clínicas e profissionais de saúde animal passam por processos de transformação digital, surge a necessidade de sistemas capazes de integrar informações, otimizar atendimentos e centralizar dados de forma segura, eficiente e acessível. Nesse contexto, a criação do Sistema Auravet representa uma resposta tecnológica completa às demandas operacionais de uma clínica contemporânea, alinhada ao propósito da marca: “cuidar é natural.”

O sistema foi desenvolvido com foco em experiência do usuário, escalabilidade e organização estrutural, utilizando uma arquitetura em monorepo que integra frontend e backend em um único ecossistema. Essa abordagem permite padronização, reutilização de componentes e manutenção simplificada, além de facilitar o desenvolvimento colaborativo.

A aplicação combina tecnologias modernas como React, Vite e TailwindCSS no frontend, Node.js, Express e Prisma ORM no backend e PostgreSQL como banco de dados, todos orquestrados por uma stack containerizada via Docker Compose. O resultado é uma plataforma robusta, modular e preparada para evoluções futuras.

A construção do sistema também reflete os valores da Auravet, traduzidos na identidade visual minimalista e acolhedora, na linguagem clara e empática e na preocupação com acessibilidade e usabilidade. A proposta digital acompanha a essência da marca, reforçando o compromisso com o bem-estar dos pets e a confiança dos tutores, ao mesmo tempo em que entrega eficiência e controle para a equipe clínica.

Esta documentação apresenta de forma detalhada todo o processo de desenvolvimento do sistema, abrangendo arquitetura, modelagem, requisitos, tecnologias empregadas, segurança, testes, diretrizes visuais, fluxos operacionais e referência completa da API. O objetivo é fornecer uma visão clara e estruturada do funcionamento da plataforma, servindo como guia técnico e como registro formal do projeto.

A partir deste ponto, serão exploradas todas as camadas do sistema Auravet, demonstrando como a tecnologia, o design e o cuidado se unem para materializar uma solução digital humanizada, sustentável e preparada para atender às necessidades reais de uma clínica veterinária moderna.

1.1. Contextualização

O setor veterinário vive um momento de transformação digital, impulsionado pela necessidade de unificar informações clínicas, melhorar o atendimento ao tutor e facilitar a rotina administrativa das equipes. Em muitas clínicas, processos essenciais – como controle de estoque, registro de atendimentos, gestão de caixa, agendamento e acompanhamento de pacientes – ainda são feitos de forma manual ou em sistemas fragmentados, dificultando a comunicação e aumentando o risco de falhas.

A Auravet, clínica veterinária digital-first movida pela premissa de que “cuidar é natural”, nasce com o propósito de unir acolhimento, tecnologia e ciência em uma jornada de cuidado completa. Esse olhar humano, aliado à inovação, inspirou o desenvolvimento de um sistema próprio que centraliza operações e simplifica o trabalho clínico e administrativo.

O Sistema Auravet organiza, integra e automatiza a gestão de tutores, pets, atendimentos, serviços, produtos, caixa, agenda e permissões internas. A plataforma foi construída com atenção ao design, usabilidade e à identidade da marca – oferecendo uma experiência leve, acolhedora e eficiente para equipes veterinárias e administrativas.

Dessa forma, o projeto representa não apenas uma solução tecnológica, mas a consolidação da essência da Auravet em um ecossistema digital moderno, funcional e preparado para evoluir conforme as necessidades da clínica e dos profissionais que a compõem.

1.2. Objetivo do Sistema

O Sistema Auravet tem como objetivo central oferecer uma plataforma integrada e moderna para apoiar o funcionamento clínico e administrativo da clínica, unindo em um único ambiente digital todos os processos essenciais do cuidado veterinário.

A solução foi concebida para facilitar o cadastro de tutores e pets, organizar atendimentos, registrar serviços e produtos utilizados, controlar o fluxo financeiro e estruturar a gestão interna por meio de funções e permissões específicas para cada colaborador.

Além de consolidar operações, o sistema busca refletir a identidade única da marca Aurauet, traduzindo em sua interface e experiência de uso a combinação entre empatia, organização e tecnologia. Dessa forma, o objetivo vai além da simples informatização: a plataforma foi projetada para fortalecer o vínculo entre profissionais, tutores e pets, garantindo precisão nos registros, eficiência no atendimento e uma jornada de cuidado mais leve, sustentável e alinhada ao propósito da clínica.

1.3. Escopo do Projeto

O escopo deste projeto contempla o desenvolvimento completo do Sistema Aurauet, incluindo todas as funcionalidades necessárias para suportar a rotina real de uma clínica veterinária digital-first.

O sistema abrange o cadastro detalhado de tutores, a criação e edição de perfis de pets vinculados aos seus responsáveis e o registro de usuários internos com informações profissionais como cargo, CRMV e biografia. Inclui também a implementação de um módulo robusto de funções, que permite controlar permissões e acesso a processos internos conforme o papel do colaborador na clínica.

No âmbito clínico, o sistema oferece o registro completo de atendimentos, possibilitando a seleção de serviços e produtos utilizados, sempre integrados de forma direta ao módulo de caixa, responsável pela geração de faturas e registro de pagamentos.

Além disso, o escopo abrange o gerenciamento de estoque por meio do módulo de produtos, permitindo acompanhar itens críticos, entradas, saídas e quantidade disponível para uso em atendimentos.

O catálogo de serviços consolida os procedimentos oferecidos pela clínica, facilitando tanto o registro de atendimentos quanto a gestão de preços e categorias.

O sistema também integra uma agenda destinada aos colaboradores e um módulo de agendamentos, permitindo organizar consultas, retornos e procedimentos futuros. Por fim, a plataforma apresenta uma página inicial com um dashboard que reúne informações essenciais, como totais cadastrados, status de serviços e indicadores relevantes para o dia a dia da clínica.

Embora o sistema seja amplo e completo, funcionalidades como prontuário veterinário avançado, aplicativo mobile, portal externo para tutores e integrações com sistemas de pagamento não fazem parte desta versão inicial, permanecendo como oportunidades de expansão futuras.

1.4. Público-Alvo

O Sistema Aurauet foi projetado para atender de forma integrada e eficiente as diferentes equipes internas da clínica, criando um ecossistema digital que facilita a rotina operacional e clínica, ao mesmo tempo em que fortalece o cuidado humanizado que define a marca. O público-alvo principal abrange administradores, veterinários, enfermeiros, atendentes e profissionais do setor financeiro, cada um desempenhando funções diretamente suportadas pelos módulos do sistema.

Os administradores utilizam o sistema para supervisionar toda a operação da clínica. Eles realizam o gerenciamento de funções internas, controlam permissões de colaboradores, acompanham o funcionamento geral dos módulos e coordenam áreas essenciais como caixa, estoque, catálogo de serviços e gestão de usuários. Para essa equipe, o sistema oferece ferramentas que garantem a governança completa das atividades e a visão estratégica do desempenho da clínica.

Os médicos veterinários representam outro público fundamental. Eles acessam diariamente informações clínicas dos pets, registram atendimentos, selecionam procedimentos e serviços aplicados, realizam lançamentos de produtos utilizados e consultam históricos completos de cada animal. Além disso, utilizam o módulo de agenda para organizar atendimentos, retornos e rotinas clínicas, garantindo fluidez no cuidado e precisão nas informações registradas.

Os enfermeiros também desempenham papel essencial no fluxo clínico e encontram no sistema uma ferramenta prática para apoiar o trabalho diário. Eles auxiliaram os veterinários em procedimentos, fazem registros de serviços de

enfermagem quando necessários, organizam informações clínicas complementares, gerenciam insumos e materiais usados nos atendimentos e auxiliam no acompanhamento pós-procedimento. Essa equipe encontra no sistema um meio de agilizar o fluxo de cuidados e manter os registros clínicos coesos e atualizados.

Os atendentes e auxiliares administrativos utilizam o sistema para registrar tutores, cadastrar novos pets, atualizar informações de cadastro, realizar e ajustar agendamentos e garantir que o fluxo de entrada de tutores e animais seja coordenado de forma clara e organizada. Eles atuam como o primeiro ponto de contato entre a clínica e o tutor, sendo apoiados por telas intuitivas, descrições claras e um fluxo de uso acolhedor, condizente com a identidade da Auravet.

Por fim, a equipe financeira, contadores e os operadores de caixa utilizam o módulo de pagamentos para gerenciar cobranças decorrentes dos atendimentos. Eles acessam as faturas geradas automaticamente a partir dos serviços e produtos registrados pelos profissionais de saúde, controlam pagamentos recebidos, atualizam status financeiros e contribuem para o fechamento de caixa diário. O sistema, portanto, oferece a esse público uma visão clara, transparente e eficiente do fluxo financeiro da clínica.

1.5. Metodologia de desenvolvimento

O desenvolvimento do Sistema Auravet foi conduzido seguindo uma abordagem iterativa e incremental, inspirada nos princípios ágeis que colocam o usuário final no centro do processo de construção.

Desde o início, adotou-se uma postura orientada à validação contínua, permitindo que o sistema evoluísse de forma orgânica, sempre incorporando aprendizados, ajustes e refinamentos ao longo de cada ciclo de desenvolvimento.

A criação do projeto teve início com a definição da identidade visual e dos valores principais da marca – empatia, cuidado e tecnologia – que serviram como diretrizes tanto para a interface quanto para a experiência de uso.

Em seguida, estruturou-se o monorepo e os componentes fundamentais de arquitetura, definindo padrões de código, camadas da aplicação, ambientes e integrações.

A partir desse alicerce técnico, o desenvolvimento avançou em sprints curtas, cada uma focada na entrega de um conjunto funcional claro e validável. Em um primeiro momento, foram estabelecidas as bases do backend, incluindo autenticação, permissões, modelagem do banco de dados e documentação inicial da API. Em paralelo, o frontend passou pela implementação dos componentes principais, como layout, navegação, tema visual e páginas base que respeitam o estilo acolhedor e minimalista da Auravet.

Com o avanço dos ciclos, funcionalidades mais complexas, como registro de atendimento, integração com o módulo de caixa, gestão de produtos com estoque e agenda de colaboradores, foram sendo construídas, testadas e estabilizadas.

Durante todo o processo, houve um cuidado contínuo com boas práticas, incluindo validação de dados, padrões de arquitetura, testes automatizados, versionamento adequado e acompanhamento constante da evolução do sistema por meio de revisões técnicas.

Essa metodologia não apenas garantiu a qualidade da solução entregue, como também proporcionou flexibilidade para ajustes ao longo do caminho, permitindo que o projeto permanecesse fiel ao propósito inicial da Auravet: oferecer uma experiência digital leve, funcional e integrada ao cuidado veterinário humanizado.

1.6. Cronograma de desenvolvimento

O desenvolvimento do Sistema Auravet teve início em 11 de outubro, marcando o começo da primeira sprint dedicada à estruturação da arquitetura, definição do monorepo e preparação dos ambientes. As semanas seguintes foram organizadas de forma a permitir entregas incrementais, criando um fluxo contínuo de evolução técnica e funcional.

Cada sprint focou em um conjunto de módulos específicos, garantindo tempo suficiente para implementação, testes, revisão e alinhamento com os objetivos do projeto.

Entre outubro e novembro, o sistema avançou rapidamente, passando pela construção dos módulos de autenticação, cadastro de tutores e pets, funções e permissões internas, catálogo de serviços, gerenciamento de estoque e módulo financeiro. À medida que cada funcionalidade era concluída, ela era integrada ao restante da aplicação, reforçando a coesão entre backend e frontend.

Já na segunda metade do ciclo, foram implementados recursos mais avançados, como o registro de atendimentos, o fluxo completo de caixa, o módulo de agenda e agendamentos, além das telas de dashboard que consolidam dados importantes na página inicial.

O desenvolvimento foi concluído em 29 de novembro, após diversas iterações que priorizaram refinamento visual, correção de inconsistências, estabilização da arquitetura e testes de ponta a ponta. O cronograma final reflete um percurso consistente, no qual cada etapa agregou maturidade ao sistema e garantiu que a plataforma estivesse totalmente funcional, estável e fiel à identidade da Auravet no momento da entrega, conforme pode ser visualizado na tabela 1.

Etapa	Descrição	Início	Fim
Planejamento e Identidade	Definição do branding Auravet, conceito, valores e diretrizes visuais	11/10	13/10
Arquitetura e Monorepo	Estruturação inicial, definição da arquitetura, organização de pastas e setup do repositório	14/10	18/10
Backend Inicial	Modelagem do banco, autenticação, permissões e base da API	19/10	25/10
Frontend Base	Implementação do layout, tema, navegação e componentes principais	26/10	01/11
Módulo de Tutores e Pets	Cadastro, edição, vínculo tutor-pet e ajustes de perfil	02/11	06/11
Módulo de Serviços e Produtos	Catálogo de serviços, estoque, itens críticos e módulo de produtos	07/11	11/11
Módulo de Atendimentos	Registro completo do atendimento, seleção de serviços e produtos	12/11	16/11
Módulo de Caixa	Faturas, pagamentos, integração com atendimentos	17/11	20/11
Agenda e Agendamentos	Agenda interna, marcação de atendimentos e organização dos profissionais	21/11	24/11

Dashboard e Refinamentos	Resumo operacional, ajustes visuais e otimizações gerais do sistema	25/11	27/11
Testes e Documentação Final	Revisão, estabilização, revisão acadêmica e entrega final	28/11	29/11

2. Desenvolvimento

A seção de Desenvolvimento apresenta a construção técnica do Sistema Auravet em profundidade, reunindo todos os elementos que sustentam sua arquitetura, funcionamento e identidade. Este capítulo revela como a aplicação foi planejada, estruturada e implementada, destacando o equilíbrio entre tecnologia, experiência do usuário e os valores de cuidado humanizado que norteiam a marca.

Primeiramente, abordamos a arquitetura geral do sistema, explicando a organização interna do monorepo, a estrutura de diretórios e a divisão em camadas que orienta a lógica de funcionamento da plataforma. Em seguida, apresentamos os principais diagramas, que ilustram visualmente sua arquitetura, seus componentes e os fluxos internos de comunicação.

Na sequência, explicamos os requisitos funcionais e não funcionais que definem o comportamento esperado do sistema e orientam as decisões de engenharia. Avançamos então para a modelagem do sistema, detalhando seus casos de uso e diagramas de sequência, que demonstram como tutores, pets, colaboradores e módulos interagem durante os diferentes processos.

Também dedicamos uma parte importante do capítulo às tecnologias utilizadas. São descritos os elementos do frontend, backend, banco de dados e infraestrutura, bem como as justificativas para cada escolha, sempre alinhadas à modernidade, escalabilidade e segurança necessárias para um sistema digital-first como a Auravet.

A identidade visual e as diretrizes de design ocupam um papel central neste capítulo, já que ajudam a traduzir para a interface a essência da marca, garantindo uma experiência leve, acolhedora e coerente entre todas as telas. Em seguida, discutimos as práticas de segurança aplicadas, incluindo autenticação, autorização e mecanismos de proteção contra vulnerabilidades comuns.

A seção também apresenta os testes automatizados que sustentam a confiabilidade do sistema, além das convenções de código, ferramentas de Linting e Hooks que asseguram padronização e qualidade durante o desenvolvimento. Por fim, o capítulo descreve o guia de deploy, a referência completa da API e as principais telas da aplicação, formando uma visão abrangente do funcionamento do Sistema Auravet.

2.1. Arquitetura do Sistema

A arquitetura do Sistema Auravet foi concebida para oferecer um ambiente de desenvolvimento moderno, estável e escalável, capaz de acomodar o crescimento natural da clínica e a expansão de funcionalidades ao longo do tempo. A organização do projeto em formato de monorepo permite que backend e frontend coexistam em uma mesma estrutura, compartilhando configurações, dependências e padrões, o que simplifica o fluxo de trabalho e torna o desenvolvimento mais eficiente. Cada parte do sistema foi planejada com foco na separação clara de responsabilidades, garantindo que cada módulo cumpra um papel específico e mantenha a coerência entre lógica, interface e infraestrutura.

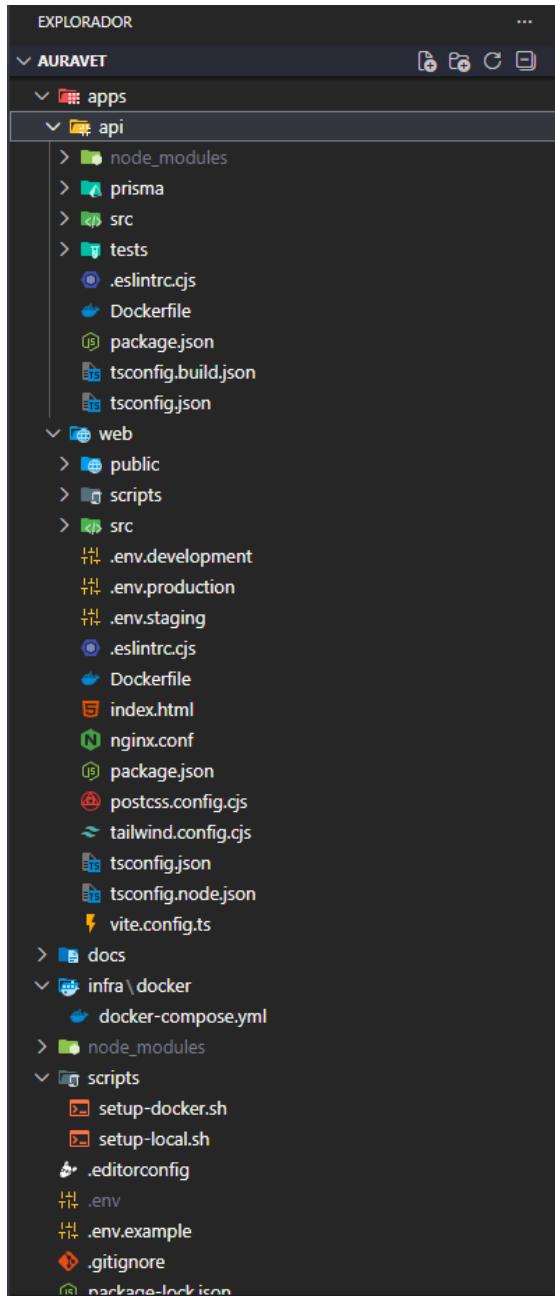
No núcleo da arquitetura, o monorepo concentra dois aplicativos principais: a API, construída com Node.js, Express e Prisma, e a interface web, desenvolvida com React, Vite e TailwindCSS. A divisão entre essas duas camadas respeita a modelagem em camadas do sistema, separando a camada de apresentação da camada de dados e das regras de negócio. Enquanto o backend é responsável por processar requisições, validar dados, interagir com o banco e expor os Endpoints da API, o frontend traduz essas operações em telas intuitivas, alinhadas ao design acolhedor da Auravet. Esta organização permite que ambos avancem de forma independente, mas sempre mantendo um contrato claro de comunicação por meio da API documentada com Swagger.

A arquitetura também contempla uma estrutura de infraestrutura isolada, localizada na pasta `infra/docker`, que permite subir o sistema completo via Docker Compose. Essa camada adicional garante que ambientes diferentes – sejam eles locais ou em servidores – possam reproduzir a mesma configuração com previsibilidade, incluindo banco de dados, API e versão final compilada do frontend. Dessa forma, a arquitetura da Auravet combina modularidade, simplicidade e uma

forte orientação a práticas profissionais, compondo um ecossistema robusto para um sistema veterinário moderno.

2.1.1. Estrutura de Diretórios

Figura 1. Diretório do projeto Auravet



A estrutura de diretórios do monorepo Auravet foi organizada para separar de forma clara as responsabilidades entre backend, frontend, infraestrutura e automações. Essa organização facilita o desenvolvimento, a manutenção e o crescimento do sistema, além de refletir a arquitetura em camadas adotada no projeto. A árvore a seguir apresenta a disposição completa dos arquivos e diretórios, permitindo visualizar como a solução foi estruturada internamente.

2.1.2. Arquitetura em camadas

A arquitetura do Sistema Auravet foi estruturada seguindo o modelo clássico de camadas horizontais, que organiza a aplicação de forma modular e facilita tanto a manutenção quanto a evolução futura do projeto. Cada camada possui responsabilidades bem definidas e se comunica apenas com as camadas diretamente adjacentes, garantindo um fluxo lógico consistente e uma separação clara entre interface, lógica de aplicação e dados.

A camada de apresentação é responsável pela interface visual da clínica e foi construída utilizando React, Vite e TailwindCSS. Ela concentra os componentes que compõem a experiência do usuário, como formulários, tabelas, dashboards, navegação e telas específicas dos módulos. Essa camada transforma as informações enviadas pela API em elementos visuais

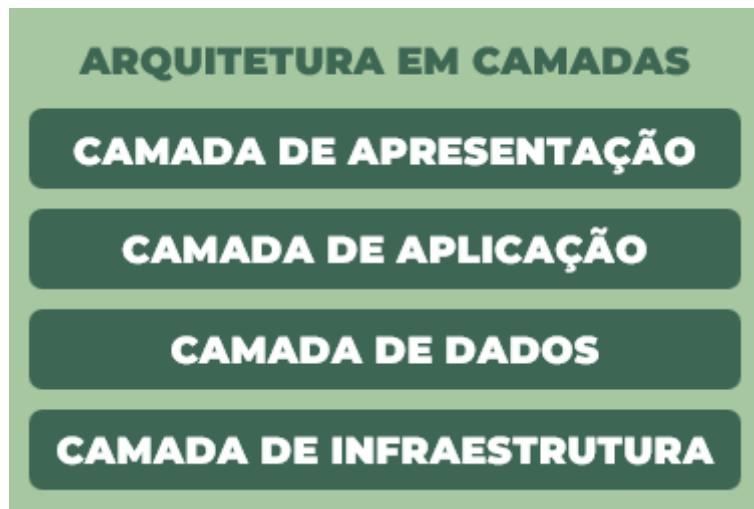
acessíveis, acolhedores e alinhados à identidade da Auravet, reforçando o propósito de cuidado natural em cada interação.

Logo abaixo, a camada de aplicação reúne toda a lógica operacional do sistema, implementada em Node.js com Express. Nessa camada são processadas as regras de negócio, verificações de permissões, validações de dados via Zod, tratamento de erros e integração entre os módulos internos. É aqui que ocorrem ações como registrar um atendimento, criar um tutor, atualizar o estoque ou gerar uma fatura no módulo de caixa. Essa camada representa o coração funcional da plataforma e atua como intermediária entre a interface e os dados persistidos.

Na base da arquitetura encontra-se a camada de dados, composta pelo banco PostgreSQL e pelo ORM Prisma, que estabelece o schema, gerencia migrações, garante consistência entre modelos e provê acesso seguro e tipado às informações. Todo o registro de tutores, pets, usuários, atendimentos, produtos, serviços, agendas e permissões é armazenado nessa camada, formando o núcleo estrutural das informações da Auravet.

Complementando esse conjunto, a camada de infraestrutura organiza a forma como o sistema é executado em ambientes diferentes, incluindo os serviços Docker, a stack Docker Compose, arquivos de build, scripts de automação e configurações que tornam o deployment mais padronizado. Essa camada garante que a aplicação possa ser rodada localmente ou em ambiente de produção com previsibilidade, estabilidade e facilidade.

Figura 2. Exemplo da arquitetura em camadas

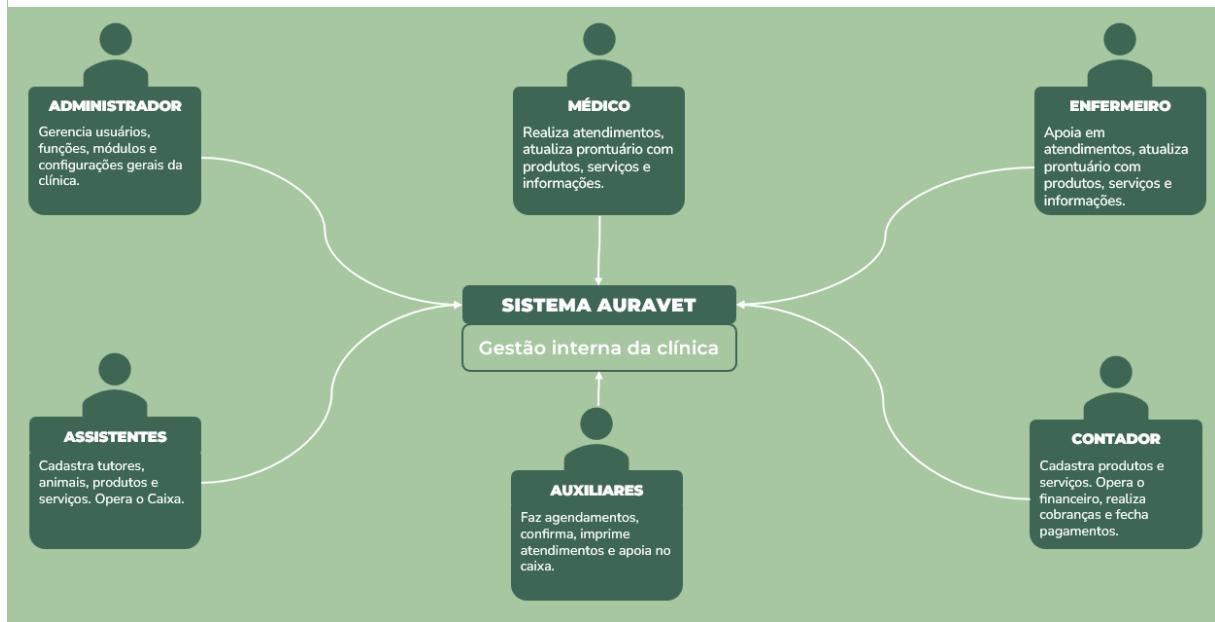


Essa organização em camadas permite que cada parte do sistema evolua de forma independente, mas sempre mantendo coerência entre si. Isso fortalece a manutenibilidade da solução e assegura que a Aurauet continue crescendo com uma base tecnológica sólida, clara e sustentável.

2.1.3. Diagramas de Arquitetura

Para representar visualmente a arquitetura do Sistema Aurauet, foram utilizados diagramas no modelo C4, que organizam a visão em diferentes níveis de abstração. No contexto deste projeto, dois níveis são suficientes para comunicar a estrutura principal: o Nível 1 (Contexto), que mostra o sistema como um todo e os usuários que interagem com ele, e o Nível 2 (Containers), que detalha como o sistema é dividido entre frontend, backend e banco de dados.

Figura 3. Diagrama C4 - Nível 1



O diagrama de contexto evidencia o Sistema AURAVET como uma solução única acessada por diferentes perfis de colaboradores da clínica, sem entrar em detalhes

de tecnologia. Já o diagrama de containers aprofunda essa visão, separando a interface web, a API e o banco de dados em blocos distintos, e indicando os principais fluxos de comunicação entre eles. Juntos, esses diagramas fornecem uma visão clara da arquitetura, facilitando o entendimento do funcionamento geral da plataforma e servindo como base para futuras evoluções.

Figura 4. Diagrama C4 - Nível 2



Além disso, foram elaborados dois diagramas complementares que ajudam a visualizar o funcionamento dinâmico do sistema: o fluxo de requisição, que demonstra o caminho percorrido por uma operação típica dentro da aplicação, e o diagrama de infraestrutura, que representa como a solução é executada em ambiente local ou containerizado. Dessa forma, a seção consolida

uma visão abrangente da estrutura do Sistema Auravet.

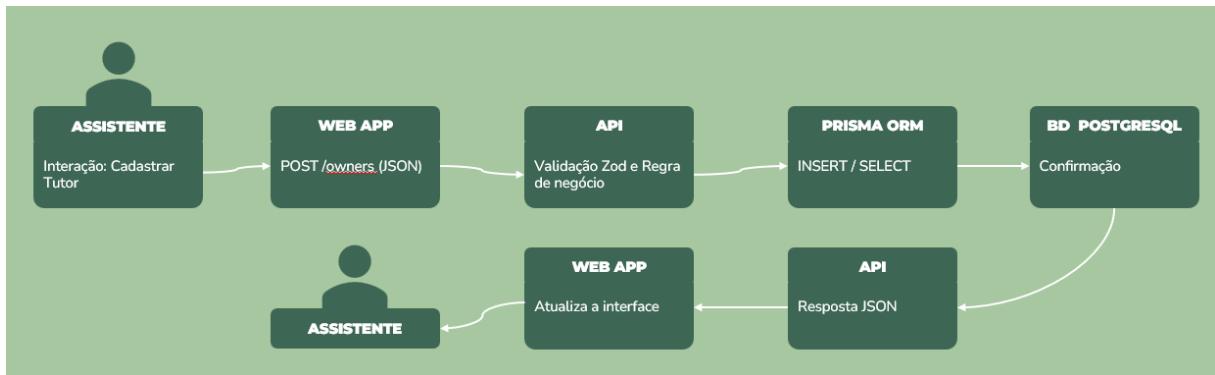


Figura 5. Fluxo de requisição – caminho de uma operação interna

Os diagramas apresentados fornecem uma visão clara e progressiva da arquitetura do Sistema Auravet. O nível de contexto posiciona o sistema dentro do ambiente da clínica; o nível de containers evidencia sua divisão estrutural interna; e os diagramas complementares de fluxo e infraestrutura mostram o comportamento operacional e o ambiente de execução.

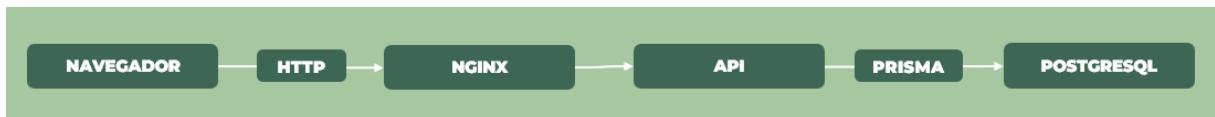


Figura 6. Diagrama de infra na execução via Docker

Em conjunto, essas representações trazem clareza ao funcionamento da aplicação, reforçando sua composição modular, escalável e alinhada ao propósito da Auravet: oferecer tecnologia que cuida.

2.2. Requisitos do sistema

Os requisitos do Sistema Auravet foram definidos de forma simulada, considerando o funcionamento ideal de uma clínica veterinária fictícia criada exclusivamente para fins educacionais. Embora a Auravet não exista como negócio real, o levantamento dos requisitos buscou representar cenários verossímeis de uma clínica moderna, integrando práticas de mercado, necessidades comuns do setor e experiências reais observadas em ambientes clínicos. Dessa forma, os requisitos

funcionais e não funcionais aqui apresentados servem como base conceitual para guiar o desenvolvimento do projeto acadêmico, permitindo exercitar tomada de decisão arquitetural, organização estrutural e implementação prática de um software de gestão.

Os requisitos foram formulados com o objetivo de permitir que o sistema simule de maneira coerente os processos centrais de uma clínica veterinária – como cadastro de tutores, pets, serviços e atendimentos – sem, no entanto, representar fielmente o funcionamento de qualquer instituição real. Assim, esta documentação consolida um conjunto de necessidades hipotéticas, mas plausíveis, para fundamentar o estudo e o desenvolvimento do Sistema Auravet.

2.2.1. Requisitos funcionais

Os requisitos funcionais descrevem as operações e funcionalidades que o sistema deve ser capaz de realizar para simular o ambiente de trabalho de uma clínica veterinária. Eles foram construídos de forma fictícia, mas inspirados em fluxos reais observados em sistemas de gestão do segmento.

ID	Requisito Funcional	Descrição
RFO1	Cadastro de tutores	O sistema deve permitir registrar novos tutores com informações essenciais.
RFO2	Cadastro e edição de pets	O sistema deve permitir criar e editar perfis de pets vinculados a tutores.
RFO3	Cadastro de colaboradores	O sistema deve permitir cadastrar usuários internos, incluindo cargo, CRMV e biografia.
RFO4	Gestão de funções	O sistema deve permitir criar e ajustar funções com permissões específicas.
RFO5	Registro de atendimento	O sistema deve simular o registro de atendimentos, serviços aplicados e produtos utilizados.
RFO6	Geração de fatura	Cada atendimento deve gerar uma fatura vinculada, simulando o controle financeiro.
RFO7	Registro de pagamentos	O sistema deve simular o registro de pagamentos associados às faturas.
RFO8	Estoque	O sistema deve simular gerenciamento de produto e controle de estoque.
RFO9	Catálogo de serviços	O sistema deve permitir cadastrar serviços clínicos e configurá-los.

ID	Requisito Funcional	Descrição
RF10	Agenda	O sistema deve simular uma agenda interna de colaboradores.
RF11	Agendamentos	O sistema deve permitir simular consultas e agendamentos futuros.
RF12	Dashboard	O sistema deve apresentar um painel inicial com contagens e indicadores básicos.
RF13	Autenticação	O sistema deve permitir login simples baseado em credenciais fictícias.
RF14	Autorização	O sistema deve restringir acesso com base em funções pré-configuradas.

2.2.2. Requisitos não funcionais

Os requisitos não funcionais estabelecem critérios de qualidade esperados em um sistema real. Eles representam boas práticas desejáveis para qualquer aplicação moderna.

ID	Requisito Não Funcional	Descrição
RNFO1	Desempenho básico	As telas devem responder de forma ágil durante a simulação de uso.
RNFO2	Segurança conceitual	A aplicação deve manter no projeto boas práticas de segurança digital, ainda que sem rigor de produção.
RNFO3	Hash de senhas	Senhas devem ser armazenadas de forma segura utilizando hashing.
RNFO4	Validação	Os dados enviados à API devem ser validados para evitar inconsistências.
RNFO5	Usabilidade	A interface deve ser intuitiva e agradável, reforçando a identidade visual da Auravet.
RNFO6	Documentação	As rotas devem ser documentadas de forma acessível.
RNFO7	Escalabilidade conceitual	A arquitetura deve permitir expansão.
RNFO8	Manutenibilidade	O código deve seguir padrões claros que facilitem leitura e estudo.
RNFO9	Portabilidade	O sistema deve ser executável tanto localmente quanto via Docker.

Os requisitos apresentados consolidam a base conceitual necessária para o desenvolvimento do Sistema Auravet, orientando tanto as funcionalidades essenciais quanto os critérios de qualidade esperados para uma aplicação moderna.

Eles estruturam de forma clara o comportamento desejado do sistema e servem como guia para a modelagem, implementação e validação das funcionalidades descritas.

Ao estabelecer esses requisitos, cria-se um cenário sólido para explorar práticas de arquitetura, desenvolvimento e organização de software, garantindo que cada módulo da Auravet opere de maneira coerente, previsível e alinhada à proposta da plataforma.

2.3. Modelagem do sistema

A modelagem do Sistema Auravet fornece uma visão estruturada das interações entre os usuários e as funcionalidades que compõem a aplicação. Por meio de casos de uso e diagramas de sequência, é possível compreender como cada colaborador se relaciona com os módulos da clínica, além de visualizar o fluxo lógico das operações internas. Esses modelos ajudam a reforçar a clareza da arquitetura, evidenciando papéis, ações principais e o comportamento esperado do sistema ao longo do seu funcionamento.

A seguir, são apresentados os principais casos de uso que representam os fluxos essenciais da plataforma, abrangendo gestão de tutores, pets, colaboradores, funções, atendimentos, produtos, serviços, agendamentos e operações financeiras. Embora simplifiquem a complexidade real do sistema, esses casos de uso oferecem uma representação fiel da experiência construída para os usuários da Auravet.

2.3.1. Casos de uso

Os casos de uso descrevem as ações mais importantes que um colaborador realiza dentro do sistema. Eles representam a relação entre os perfis de usuário e os módulos principais, permitindo visualizar, de forma objetiva, quem executa cada operação e qual o comportamento esperado em cada uma delas.

Visão geral do Sistema Auravet

O diagrama de casos de uso geral do Sistema Auravet apresenta, de forma integrada, as principais funcionalidades disponibilizadas aos diferentes perfis de usuário. Médicos e enfermeiros utilizam o sistema para verificar agenda, gerenciar atendimentos e registrar atendimentos, garantindo o registro clínico dos pets.

O assistente e o auxiliar administrativo atuam na criação de tutores e pets, criação e gerenciamento de agendamentos e no apoio operacional ao fluxo da clínica. O contador é responsável por gerenciar faturas, criar produtos e serviços e gerenciar o financeiro, assegurando o controle econômico das operações.

O administrador do sistema concentra as atividades de gerenciamento de usuários e gerenciamento de funções, definindo permissões e acesso aos módulos. Em conjunto, esses casos de uso representam a visão macro do sistema e como cada papel interage com os módulos clínicos, administrativos e financeiros.

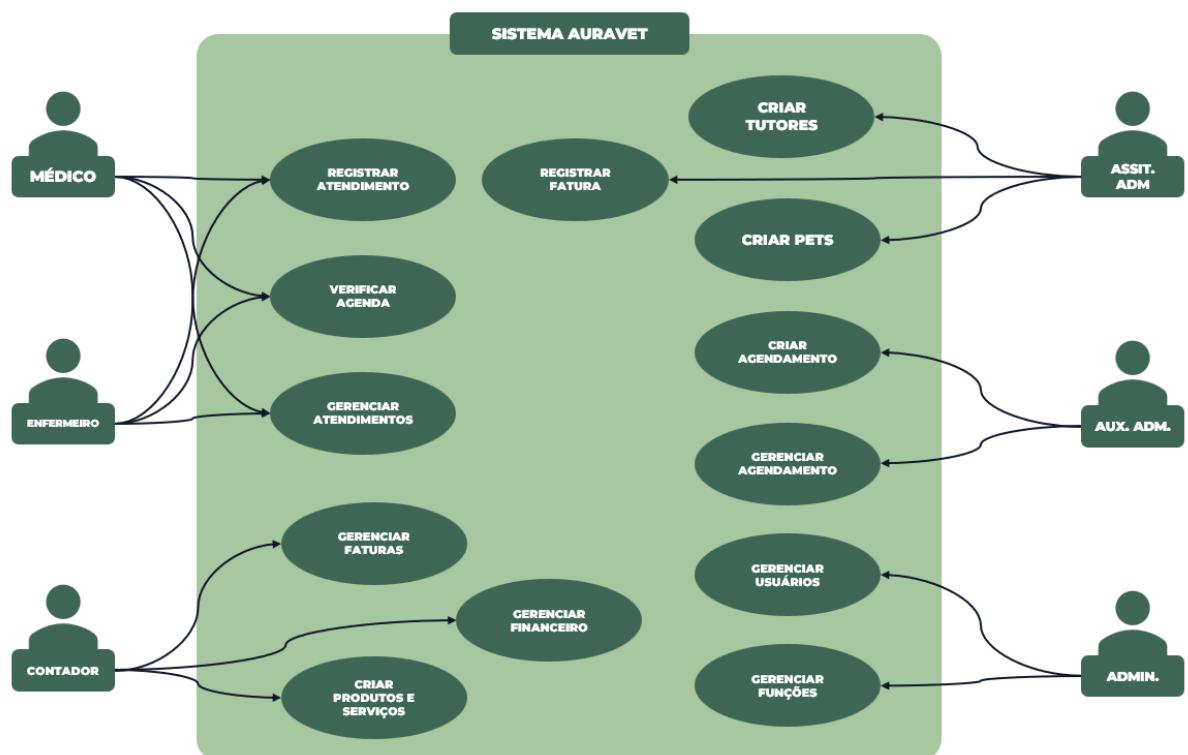


Figura 7. Diagrama de casos de uso geral do sistema.

Módulo de Agendamentos

O diagrama de casos de uso do módulo Agendamentos descreve como o sistema auxilia na organização da rotina da clínica. O auxiliar administrativo é o principal

responsável por criar agendamentos, registrando data, horário, profissional responsável e o pet atendido.

O mesmo perfil executa a função de confirmar presença, entrando em contato com o cliente previamente e atualizando o status do agendamento no sistema. Quando necessário, o auxiliar realiza o caso de uso reagendar os atendimentos, ajustando a agenda conforme solicitações dos tutores ou conflitos de horário. Médicos e enfermeiros utilizam o caso de uso visualizar agendamentos para acompanhar sua agenda diária, enquanto o administrador pode ter acesso para fins de supervisão. Esse módulo garante que a agenda da clínica seja centralizada, atualizada e visível para todos os atores envolvidos.

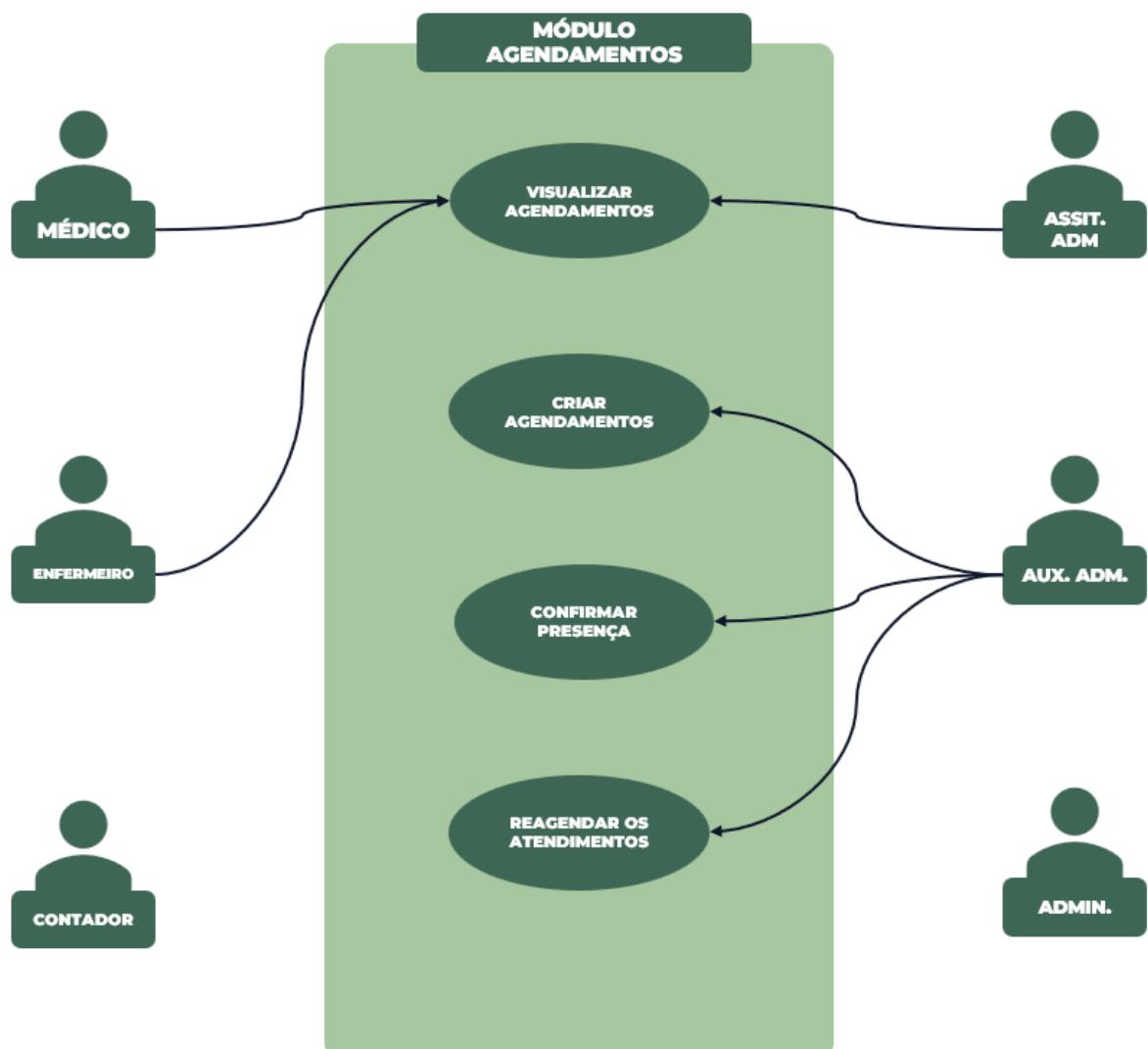


Figura 8. Diagrama de casos de uso para agendamentos

- Visualizar agendamentos – permite que médicos, enfermeiros e equipe administrativa consultem a agenda filtrando por data, profissional ou status.
- Criar agendamentos – registra novos compromissos para um pet, vinculados ao tutor e ao profissional responsável.
- Confirmar presença – registra a confirmação do cliente antes da consulta, reduzindo faltas e melhorando o uso da agenda.
- Reagendar os atendimentos – ajusta data e horário de um agendamento existente, mantendo o histórico e evitando conflitos.

Módulo Atendimento

No módulo Atendimento, o diagrama de casos de uso mostra como médicos e enfermeiros conduzem o registro clínico dos pets. O caso de uso visualizar o atendimento permite consultar atendimentos já realizados ou em andamento, com histórico de serviços e produtos utilizados.

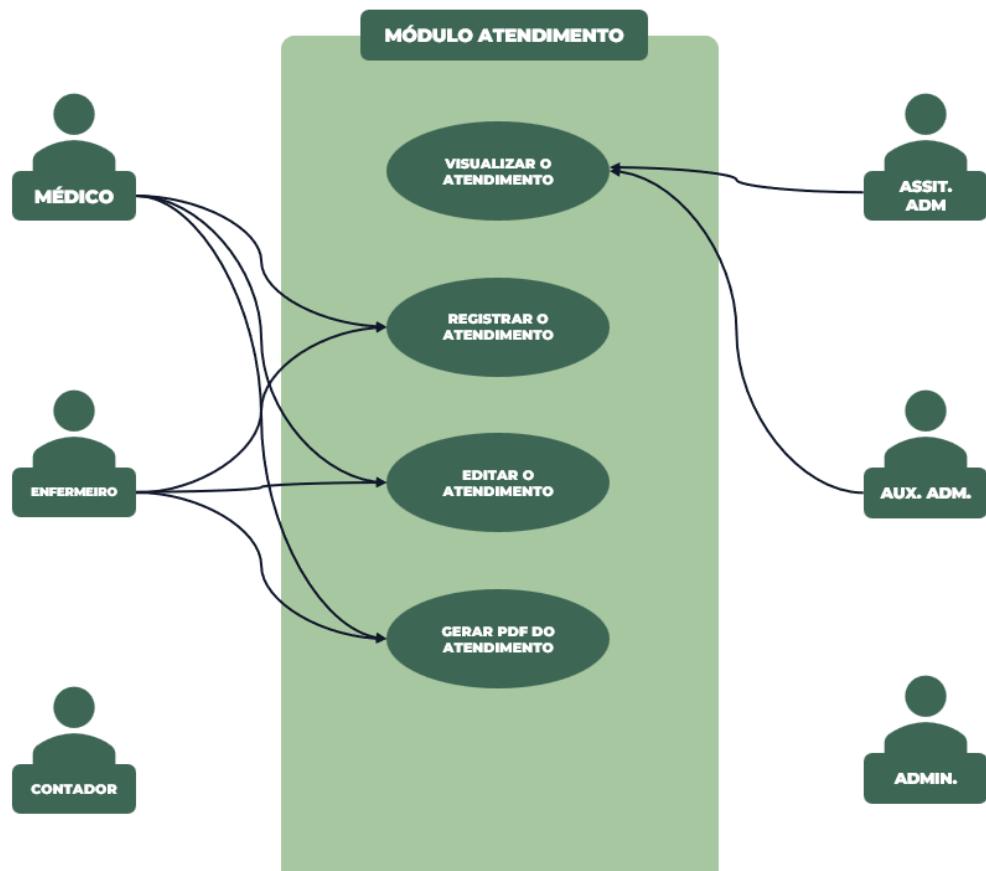


Figura 9. Diagrama de casos de uso para Atendimento

O caso de uso registrar o atendimento contempla o preenchimento dos dados clínicos principais, seleção dos serviços realizados e produtos aplicados ao pet. Quando alguma informação precisa ser alterada, o caso de uso editar o atendimento possibilita correções, mantendo a rastreabilidade.

Por fim, gerar PDF do atendimento cria um documento formal com o resumo do prontuário daquele atendimento, que pode ser impresso ou compartilhado com o tutor conforme necessidade. A equipe administrativa pode ter acesso de leitura para apoio operacional, enquanto o contador e o administrador utilizam essas informações em relatórios e conferências.

Módulo Caixa/Faturas

O diagrama de casos de uso do Módulo Caixa representa as atividades relacionadas ao faturamento e recebimento. Os perfis de assistente administrativo, auxiliar administrativo e contador acessam o caso de uso visualizar fatura para consultar os dados financeiros gerados a partir dos atendimentos.

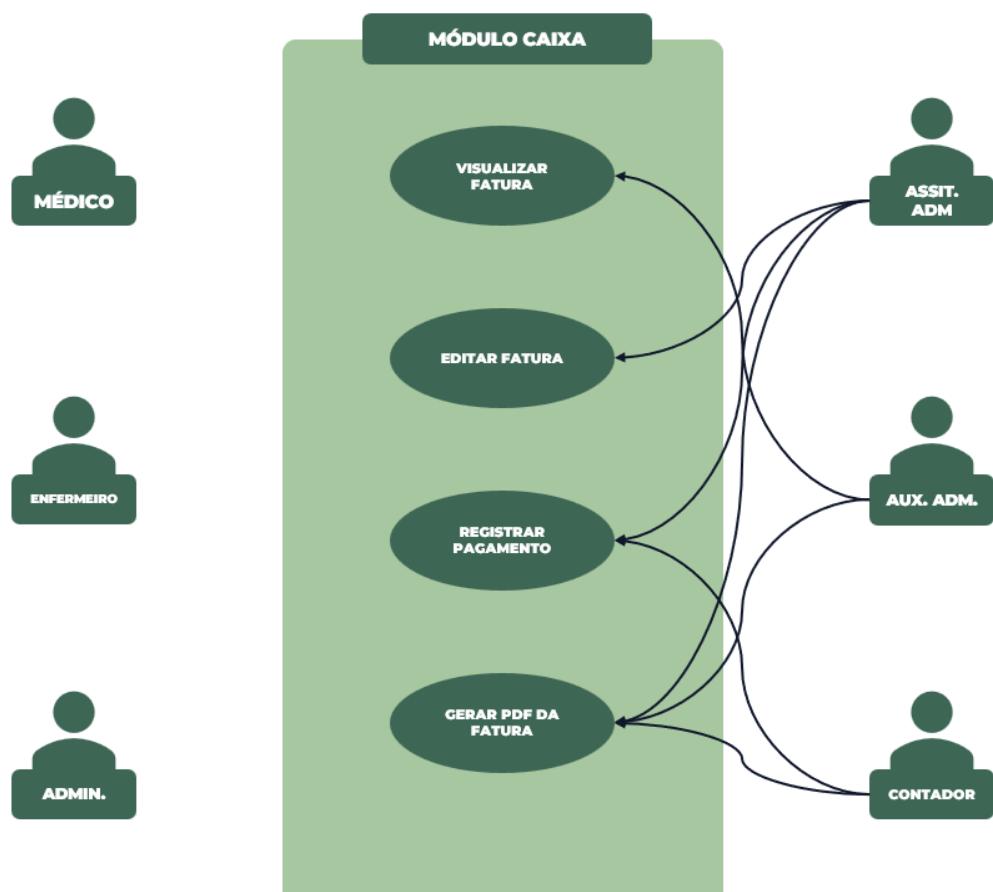


Figura 10. Diagrama de casos de uso para caixa

O caso de uso editar fatura permite ajustar informações como itens cobrados, condição de pagamento e observações acordadas com o cliente. Em seguida, o caso de uso registrar pagamento registra a quitação da fatura, armazenando dados como forma de pagamento e data de recebimento.

Por fim, gerar PDF da fatura produz o documento fiscal/recibo em formato eletrônico, que pode ser impresso e entregue ao tutor. Médicos e enfermeiros podem apenas visualizar faturas relacionadas aos seus atendimentos, enquanto o administrador mantém acesso de supervisão.

Casos de uso:

- **Visualizar fatura** – consulta das faturas por tutor, data, status ou número.
- **Editar fatura** – inclusão/ajuste de itens e atualização de dados de cobrança.
- **Registrar pagamento** – gravação do pagamento da fatura, alterando seu status.
- **Gerar PDF da fatura** – emissão de documento em PDF para impressão ou envio ao cliente.

Módulo Serviços

No módulo Serviços, o sistema concentra o cadastro e o uso dos serviços prestados pela clínica (consultas, exames, vacinas, cirurgias etc.). O caso de uso visualizar serviços permite que médicos, enfermeiros e equipe administrativa consultem a lista de serviços disponíveis, com descrição e valores sugeridos.

O caso de uso inserir serviços no atendimento é utilizado durante o registro do atendimento para vincular os serviços realizados ao pet, influenciando diretamente o valor da fatura. Já os casos de uso criar serviços e editar serviços são executados por perfis administrativos (como contador, auxiliar e administrador), responsáveis por manter a tabela de serviços atualizada, incluindo novos procedimentos ou ajustando descrições e preços conforme a realidade da clínica.

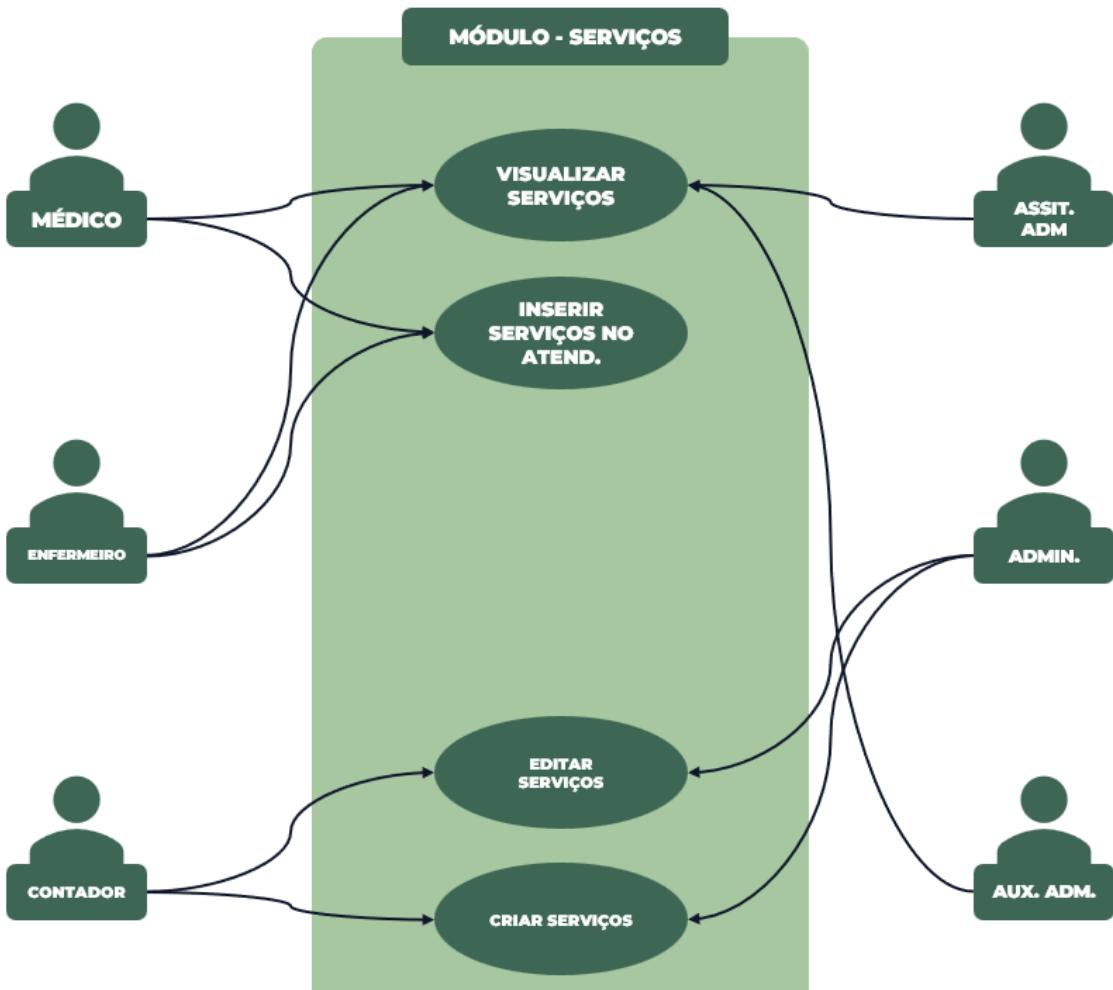


Figura 11. Diagrama de casos de uso para serviços

Módulo Produtos

O diagrama de casos de uso do módulo Produtos descreve como o sistema gerencia itens físicos, como medicamentos, materiais e produtos de venda direta. O caso de uso visualizar produtos permite que todos os perfis autorizados consultem o catálogo de produtos com informações de preço e disponibilidade.

Durante o atendimento, médicos e enfermeiros utilizam inserir produtos no atendimento para registrar medicamentos e insumos utilizados, associando-os ao prontuário e à futura cobrança. Em situações de venda direta no caixa, o caso de uso inserir produtos na fatura possibilita adicionar itens adicionais à cobrança do cliente.

O caso de uso ajustar estoque é utilizado, principalmente pelo auxiliar administrativo, para registrar entradas e saídas de produtos, mantendo o saldo

atualizado. Por fim, criar produtos e editar produtos ficam a cargo do contador e do administrador, que mantêm o cadastro oficial de itens comercializados pela clínica.

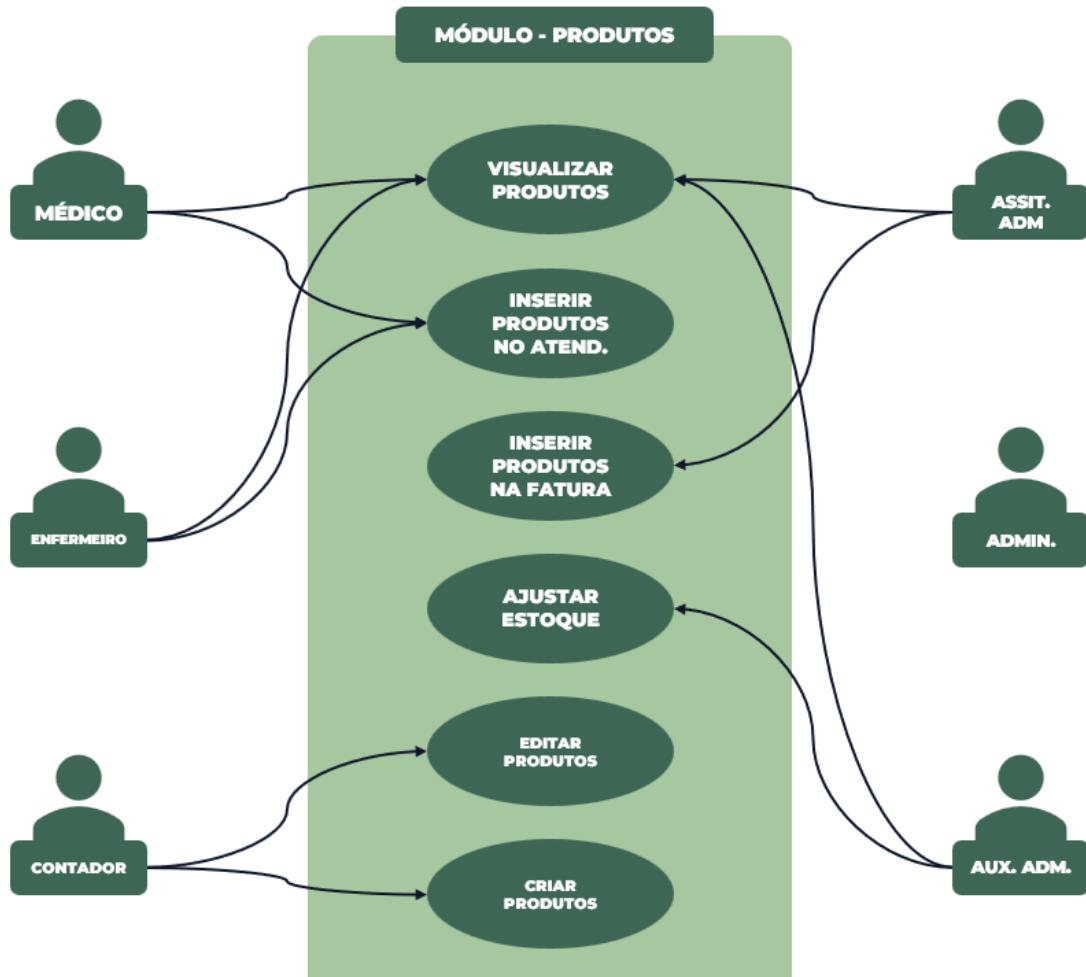


Figura 12. Diagrama de casos de uso para produtos

Módulo Pets

O módulo Pets concentra as informações dos animais atendidos. O caso de uso criar pets permite ao assistente administrativo cadastrar um novo pet vinculado a um tutor, registrando dados básicos como nome, espécie, raça e data de nascimento. O caso de uso editar pets possibilita atualizar essas informações quando necessário.

Já visualizar pets oferece uma lista ou busca por animais cadastrados, utilizada tanto pela equipe administrativa quanto por médicos e enfermeiros antes de um atendimento. Os casos de uso visualizar histórico dos pets e visualizar atendimentos dos pets permitem acessar o histórico clínico e a cronologia de atendimentos,

auxiliando na tomada de decisão médica. Por fim, exportar pets para Excel gera uma extração das informações cadastrais e históricas, útil para relatórios gerenciais e análises externas.

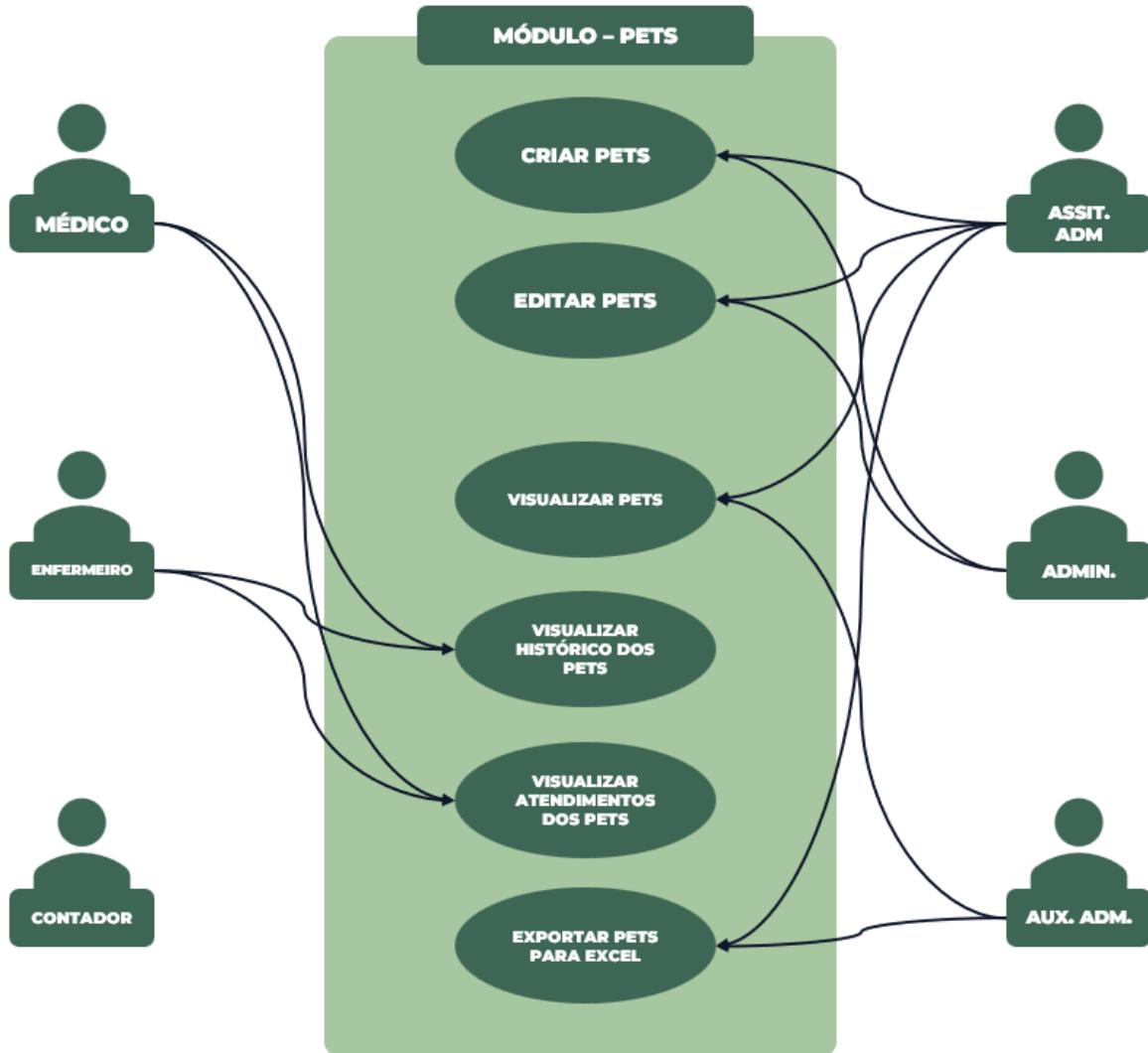


Figura 13. Diagrama de casos de uso para Pets

Módulo Tutores

No módulo Tutores, o sistema gerencia os dados dos responsáveis pelos pets. O caso de uso criar tutores permite o cadastro de novos clientes com seus dados pessoais e de contato, geralmente realizado pelo assistente administrativo no primeiro atendimento.

O caso de uso editar tutores é utilizado para manter esses dados atualizados (mudança de telefone, endereço etc.). O caso de uso visualizar tutores oferece uma

visão geral da base de clientes, com filtros por nome, documento ou outros critérios. Além disso, exportar tutores para Excel viabiliza a geração de relatórios externos e ações de relacionamento, como campanhas e comunicações específicas. Médicos, enfermeiros e o contador podem consultar essas informações para obter contexto clínico e financeiro dos atendimentos.

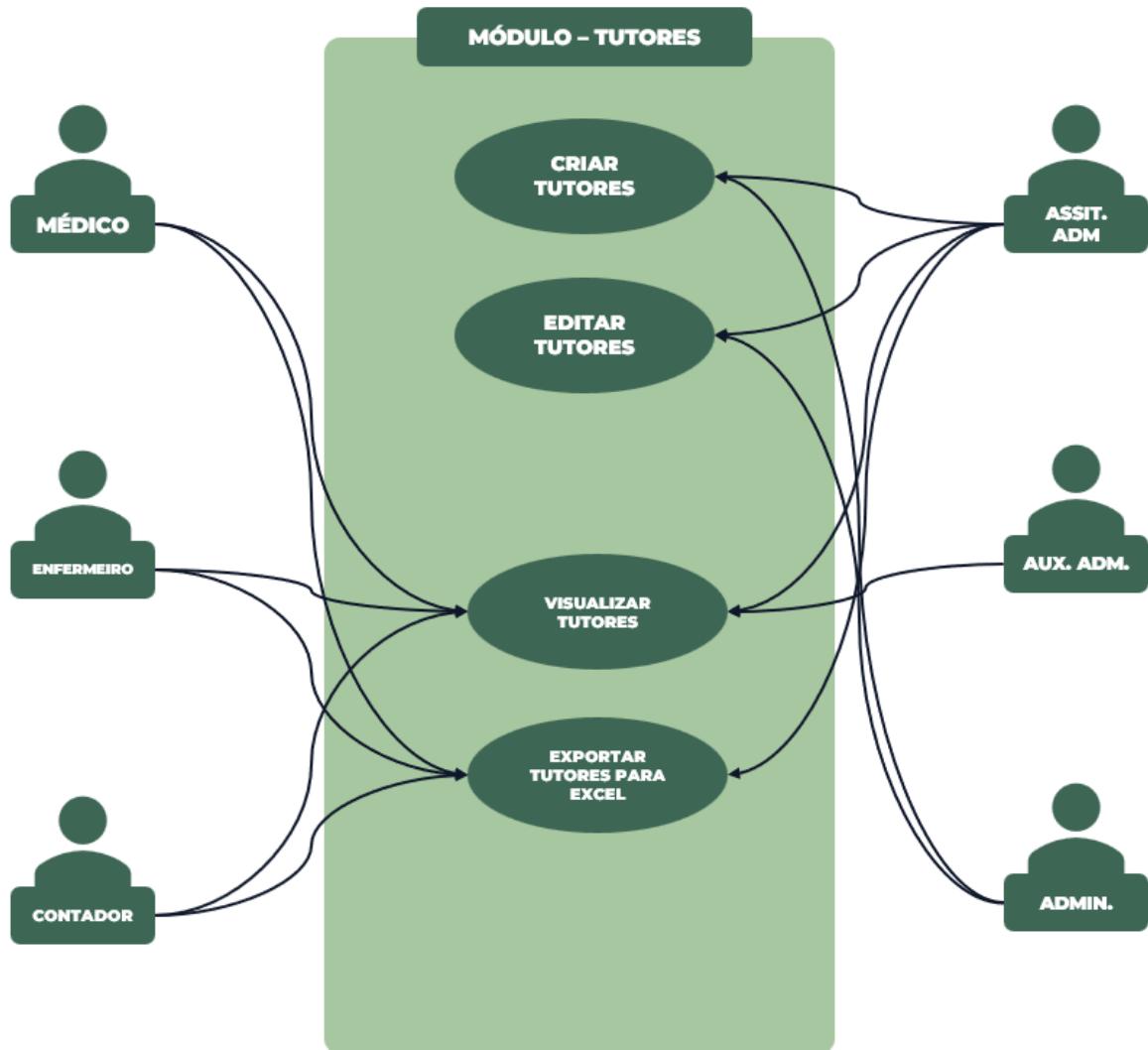


Figura 14. Diagrama de casos de uso para Tutores

Módulo Usuários

O módulo Usuários é responsável pela gestão das contas de acesso ao sistema. O caso de uso visualizar usuários permite ao administrador e aos perfis autorizados consultar a lista de usuários cadastrados, com suas funções e status de acesso. O caso de uso criar usuários é utilizado para registrar novos colaboradores, definindo seus dados de login e a função (role) associada. Quando há alteração de equipe ou

mudança de função, o caso de uso editar usuários permite atualizar permissões, dados de contato ou status de ativação. Os casos de uso inserir usuários em agendamentos e inserir usuários em atendimentos refletem o vínculo operacional entre usuários e processos: definem quais profissionais aparecem nas agendas e quais podem ser atribuídos aos atendimentos, garantindo que apenas colaboradores habilitados sejam selecionados nessas telas.

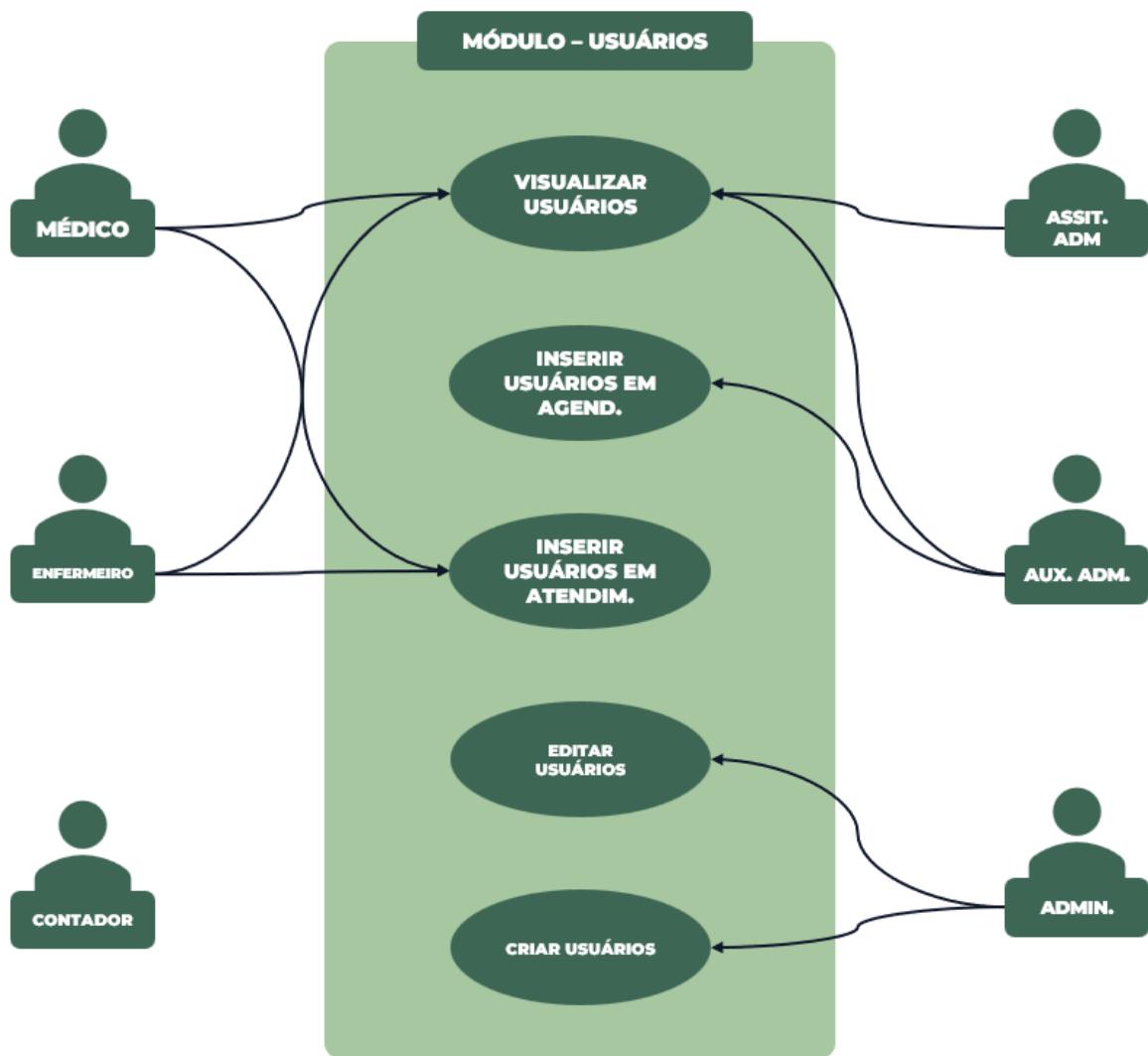


Figura 15. Diagrama de casos de uso para usuários

Módulo Funções (Roles e Módulos)

O diagrama de casos de uso do módulo Funções relaciona-se diretamente ao controle de permissões. O caso de uso visualizar funções permite ao administrador e aos perfis responsáveis por TI ou gestão de acesso consultar as funções existentes

(por exemplo, médico, enfermeiro, auxiliar, assistente, contador, administrador). O caso de uso criar funções permite definir novas funções com base nas necessidades da clínica, estruturando o modelo de papéis.

Já o caso de uso ajustar módulos nas funções está ligado ao vínculo entre funções e módulos do sistema: aqui o administrador determina quais módulos e funcionalidades cada função pode acessar (agenda, atendimentos, caixa, cadastros etc.). Esse módulo garante a aplicação do princípio de mínimo privilégio e a segurança no uso do sistema.

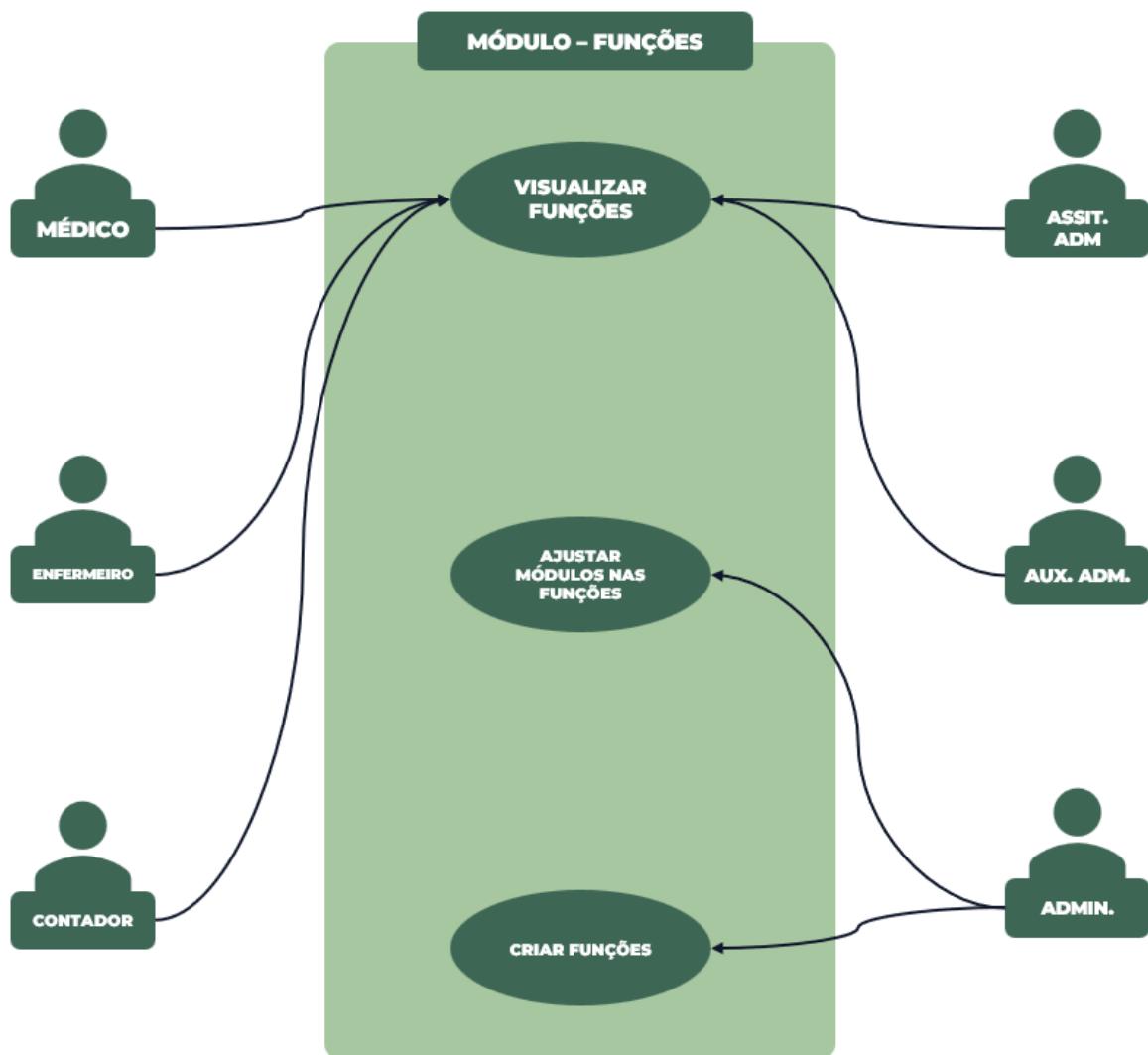


Figura 16. Diagrama de casos de uso para funções

Módulo Financeiro

O diagrama de casos de uso do módulo Financeiro representa as funcionalidades relacionadas ao acompanhamento e controle das faturas geradas a partir dos atendimentos. Os perfis de assistente administrativo, auxiliar administrativo e contador utilizam o caso de uso monitorar faturas para acompanhar a situação de cobrança dos clientes, consultando faturas abertas, vencidas ou pagas.

Quando é necessário ajustar a forma de pagamento combinada com o cliente, esses mesmos atores acessam o caso de uso ajustar informações de pagamento, atualizando dados como forma de pagamento, condição de pagamento e observações de negociação. O caso de uso registrar pagamento é responsável por registrar a quitação de uma fatura, atribuindo data de pagamento, usuário responsável e atualização do status financeiro.

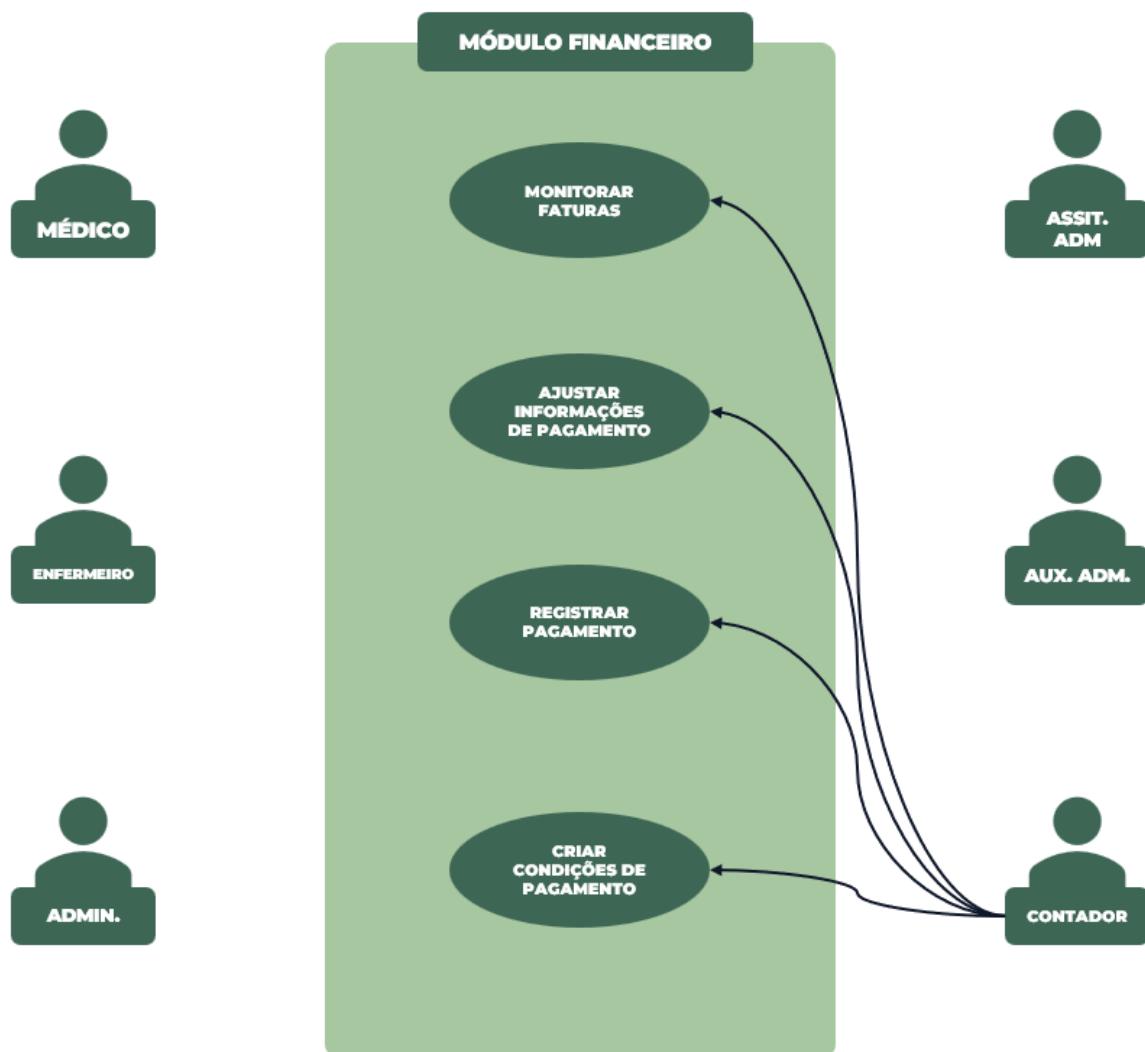


Figura 17. Diagrama de casos de uso financeiro

Por fim, o caso de uso criar condições de pagamento, executado principalmente pelo contador, permite cadastrar e manter as diferentes condições de pagamento aceitas pela clínica (à vista, parcelado, prazos em dias, entre outras), que serão utilizadas posteriormente no módulo de Caixa e no próprio Financeiro. Esse módulo garante a visibilidade da situação financeira da clínica e o alinhamento entre a cobrança efetuada e os acordos firmados com os tutores.

2.3.2. Diagrama de classe conceitual

O primeiro diagrama de classes conceitual apresenta os cadastros principais do sistema Auravet. A classe Tutor representa o responsável financeiro e legal pelos animais, armazenando dados básicos de identificação e contato. Cada tutor pode possuir um ou mais Pets, modelados pela classe Pet, que registra informações como nome, espécie, raça e data de nascimento.

As classes Produto e Servico representam, respectivamente, os itens físicos comercializados (medicamentos, materiais, produtos de venda) e os procedimentos oferecidos pela clínica (consultas, exames, vacinas, cirurgias). Por fim, a classe CondicaoPagamento descreve as formas de parcelamento e prazos utilizados nas faturas. Este diagrama foca nos cadastros que dão suporte aos processos de agendamento, atendimento e faturamento.

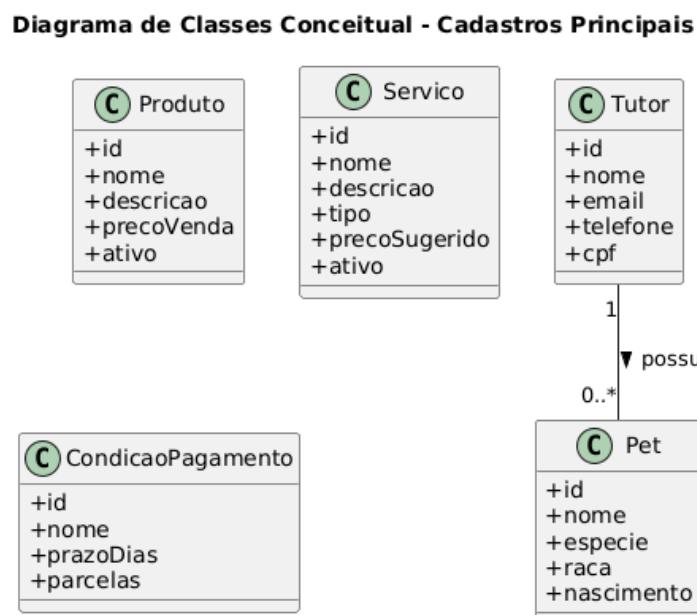


Figura 18. Diagrama de classes dos cadastros

O segundo diagrama de classes conceitual descreve o fluxo clínico do Auravet, desde o agendamento até o registro do atendimento. A classe Agendamento representa os compromissos marcados para um determinado Pet e Tutor, vinculados a um profissional da clínica (Usuario) e contendo informações de data, horário e status (agendado, confirmado, concluído etc.). A partir de um agendamento pode ser originado um Atendimento, modelado pela classe homônima, que registra as informações clínicas gerais, o valor total e o profissional responsável.

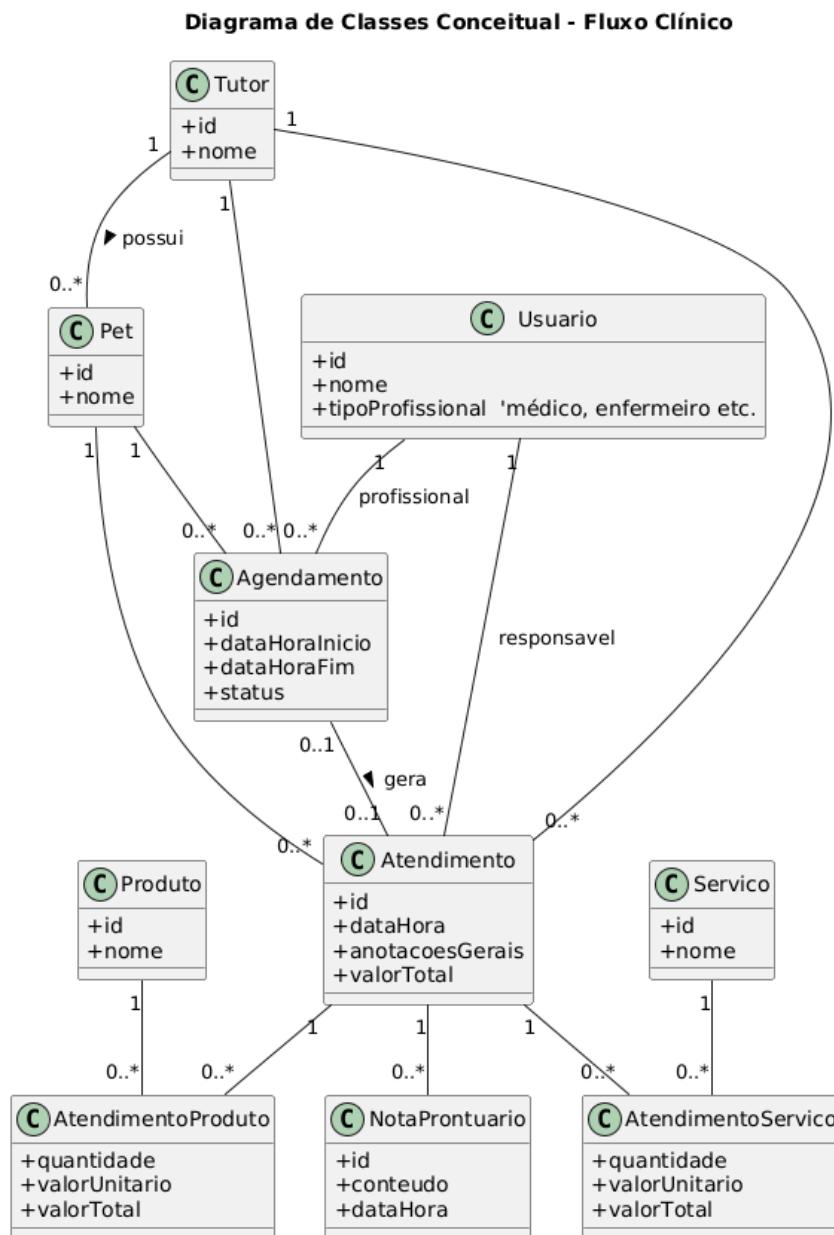


Figura 19. Diagrama de classes conceitual do Fluxo Clínico

Cada atendimento pode ter diversas Notas de Prontuário, permitindo registrar a evolução clínica e observações detalhadas. Os procedimentos realizados são representados pela associação AtendimentoServiço, que liga o atendimento aos Serviços cadastrados, enquanto os medicamentos e materiais utilizados são representados por AtendimentoProduto, que relaciona o atendimento aos Produtos. Assim, este diagrama evidencia como tutores, pets, profissionais, serviços e produtos se conectam na execução do atendimento clínico.

O terceiro diagrama de classes conceitual concentra o módulo financeiro e o controle de acesso do sistema. A classe Fatura representa a cobrança emitida ao Tutor, contendo informações de datas, valor total e observações.

Diagrama de Classes Conceitual - Financeiro e Acesso

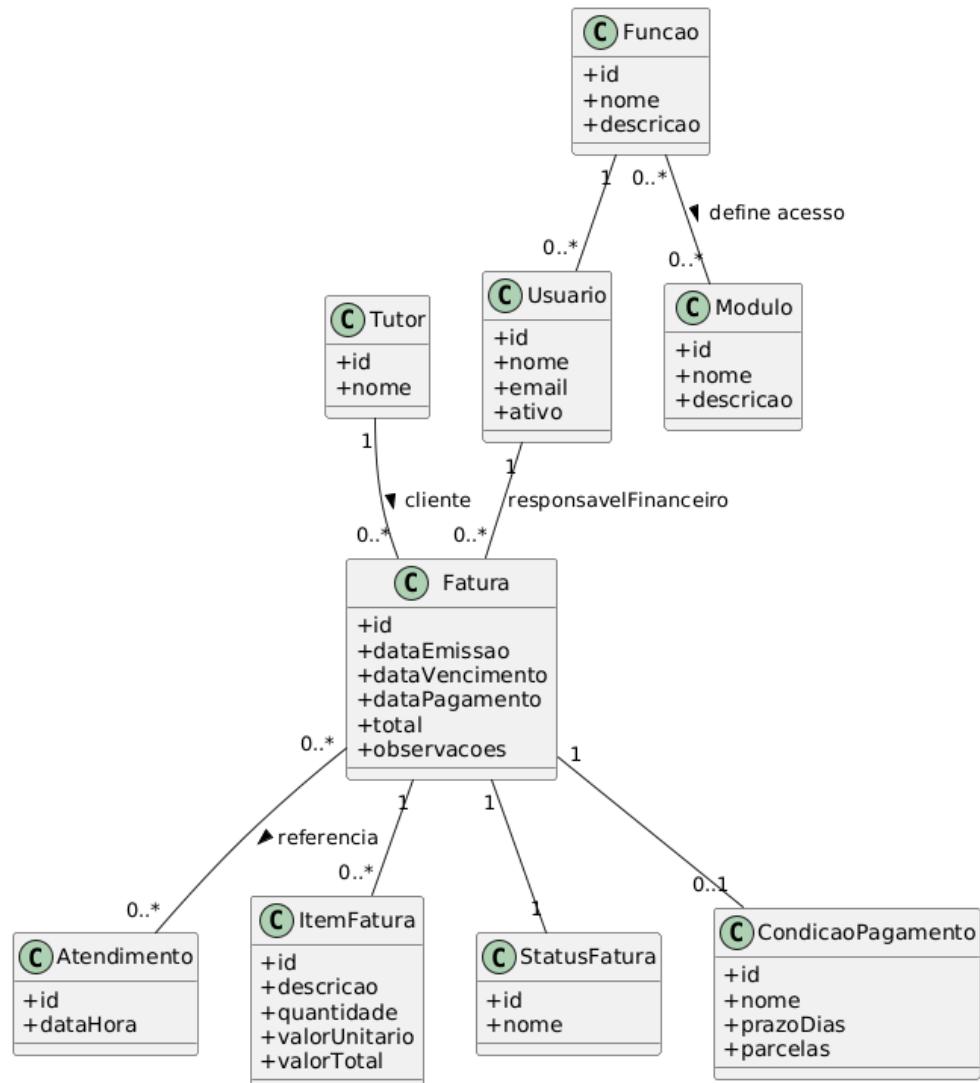


Figura 20. Diagrama de classes financeiro e acessos

Cada fatura é composta por vários Itens de Fatura, que detalham serviços e produtos cobrados. A classe StatusFatura indica a situação atual da cobrança (por exemplo, aberta, paga, vencida), enquanto a CondicaoPagamento registra o acordo estabelecido com o cliente em termos de prazo e parcelamento.

Faturas podem referenciar um ou mais Atendimentos, permitindo rastrear a origem clínica de cada cobrança, e o Usuario responsável pelo financeiro é associado à fatura como responsável pelo registro do pagamento.

Na mesma figura, o controle de acesso é modelado pelas classes Usuario, Funcao e Modulo. Cada usuário está associado a uma função (como médico, enfermeiro, auxiliar, assistente, contador ou administrador), e as funções determinam, de forma conceitual, quais módulos do sistema podem ser acessados. Dessa forma, o diagrama demonstra como a camada financeira se integra com os atendimentos clínicos e como a segurança é estruturada a partir de perfis de acesso.

2.3.3. Diagramas de atividades

Nesta subseção são apresentados os diagramas de atividades do sistema Auravet, utilizados para descrever, em nível de processo, o fluxo de ações realizadas pelos diferentes atores e pelo próprio sistema. Os diagramas evidenciam a sequência de passos, decisões e interações entre usuários e funcionalidades, permitindo visualizar de forma clara como o trabalho acontece na prática dentro da clínica veterinária.

Para este projeto foram modeladas as atividades consideradas mais críticas para o funcionamento da aplicação: Gerenciar Agendamentos, Gerenciar Atendimentos, Gerenciar Faturas (Caixa), Gerenciar Financeiro e Gerenciar Cadastros. Em conjunto, esses cinco diagramas cobrem o ciclo completo desde o contato inicial do cliente, passando pelo atendimento clínico e faturamento, até o acompanhamento financeiro e a manutenção dos cadastros que dão suporte a todo o fluxo operacional.

Gerenciar Agendamentos

O diagrama de atividades “Gerenciar Agendamentos” descreve o fluxo operacional executado principalmente pelo auxiliar administrativo. O processo

inicia quando o cliente entra em contato com a clínica e o colaborador coleta os dados do tutor e do pet. Em seguida, o auxiliar acessa a Agenda Inteligente, verifica a disponibilidade de médicos e assistentes de acordo com a escala e negocia com o cliente a data e o horário mais adequados. Ao confirmar esses dados, o sistema registra um novo agendamento com status inicial de AGENDADO.

Em uma rotina diária, o auxiliar lista os agendamentos do dia seguinte, entra em contato com cada tutor e atualiza o status para CONFIRMADO quando a presença é validada. Caso o cliente solicite mudança de horário, o colaborador verifica a disponibilidade na agenda, ajusta a data e o horário e registra o reagendamento. Esse fluxo garante que a agenda da clínica seja mantida organizada, com acompanhamento ativo de confirmações e reagendamentos.

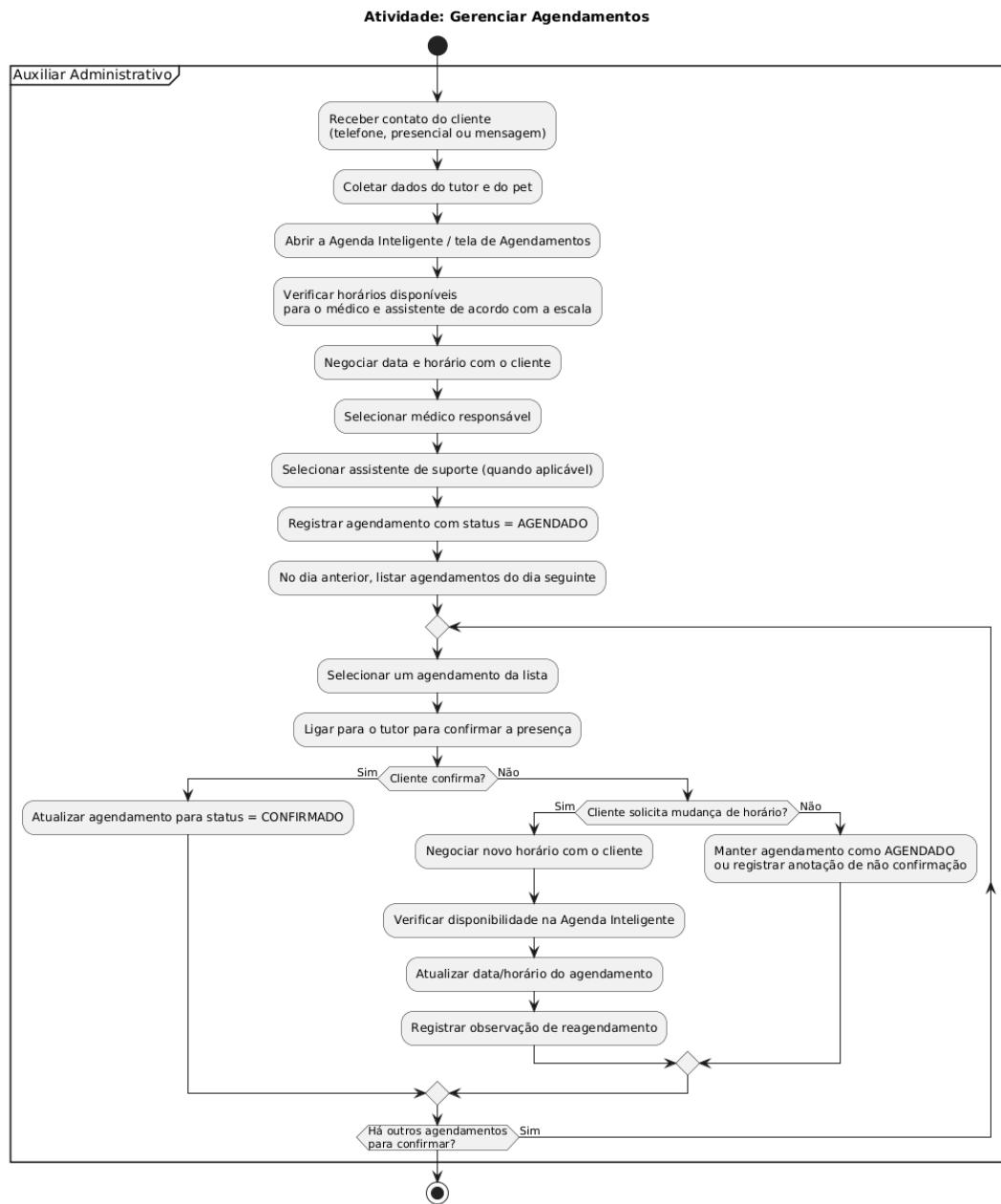


Figura 21. Diagrama de atividades para agendamentos

Gerenciar Atendimentos

O diagrama de atividades “Gerenciar Atendimentos” apresenta o fluxo executado por médicos e enfermeiros durante o atendimento clínico. A partir da Agenda Inteligente, o profissional seleciona o agendamento do pet que chegou à clínica e abre a tela de atendimento correspondente. Nesse momento são conferidos os dados do tutor e do pet e registradas as informações clínicas iniciais, como queixa principal e anamnese.

Em seguida, o profissional registra os serviços realizados (consultas, exames, vacinas, cirurgias) e os produtos utilizados (medicamentos e materiais), selecionando-os a partir dos cadastros existentes. Ao longo do processo, podem ser adicionadas notas de prontuário, que documentam a evolução e as recomendações clínicas. Após revisar o atendimento, o profissional o conclui no sistema. Automaticamente, o sistema atualiza o status do agendamento para CONCLUÍDO, calcula o valor total e gera uma fatura no módulo Caixa, estabelecendo o vínculo entre o atendimento clínico e o processo de cobrança.

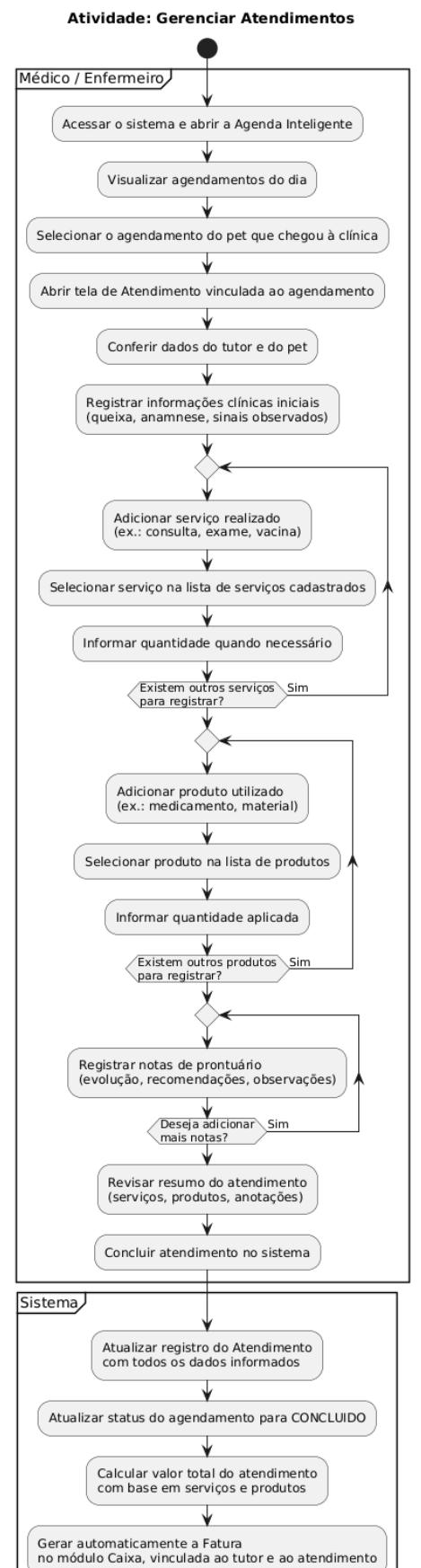


Figura 22. Diagramas de atividades do atendimento

Gerenciar Faturas

O diagrama de atividades “Gerenciar Faturas (Caixa)” representa o fluxo de trabalho do assistente administrativo e do auxiliar administrativo quando o cliente passa pelo caixa. Ao acessar o módulo Caixa, o colaborador lista as faturas abertas e seleciona aquela vinculada ao atendimento recém-finalizado. Em seguida, apresenta ao tutor o resumo de serviços e produtos cobrados. Caso o cliente deseje adquirir produtos adicionais de última hora, o colaborador os adiciona à fatura diretamente nessa tela.

Após alinhar com o cliente a forma e a condição de pagamento, o usuário atualiza os campos correspondentes, podendo incluir observações relevantes. O sistema permite a geração de um PDF da fatura, que pode ser impresso ou enviado ao tutor. Por fim, o colaborador registra o pagamento recebido, e o sistema atualiza o status da fatura para paga, armazenando a data e o responsável pelo registro. Esse fluxo garante a integração entre os atendimentos clínicos e o recebimento dos valores devidos.

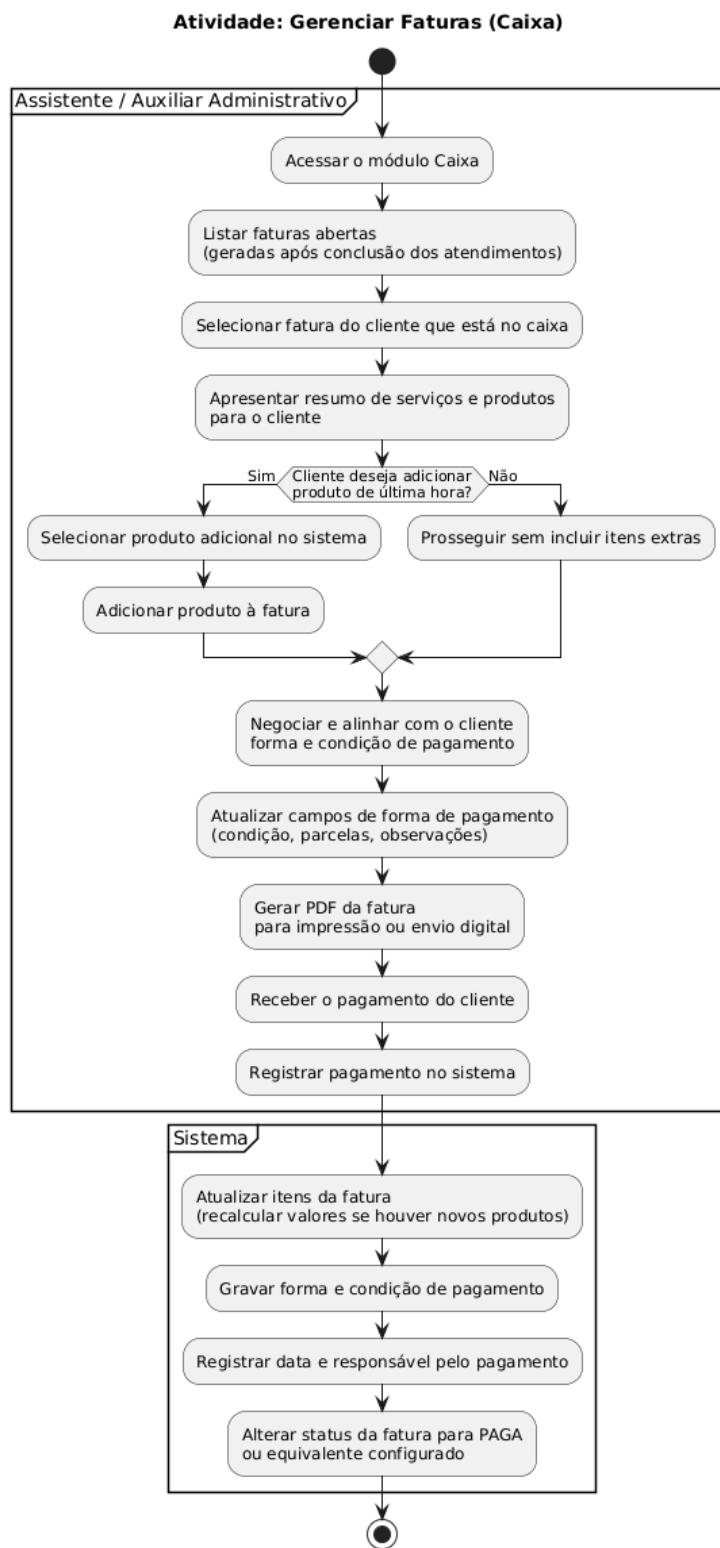


Figura 23. Diagrama de atividades para gerenciar faturas

Gerenciar Financeiro

O diagrama de atividades “Gerenciar Financeiro” aborda o trabalho do contador e do setor financeiro no acompanhamento das faturas após o atendimento. O processo inicia com o acesso ao módulo Financeiro, onde são listadas faturas por período e status (abertas, vencidas ou pagas). O contador filtra principalmente as faturas em atraso ou com maior risco de inadimplência e, para cada uma, analisa o histórico do cliente e do atendimento correspondente.

Quando necessário, o setor financeiro entra em contato com o tutor para negociar prazos ou condições de pagamento, registrando no sistema qualquer ajuste realizado. À medida que os pagamentos são confirmados, o usuário registra a quitação, e o sistema atualiza o status da fatura, mantendo o histórico de alterações e o usuário responsável pelo lançamento. Além disso, o módulo permite cadastrar novas condições de pagamento padrão, que serão usadas pelos demais módulos. Esse fluxo consolida o controle da carteira de recebíveis e oferece suporte à tomada de decisão financeira da clínica.

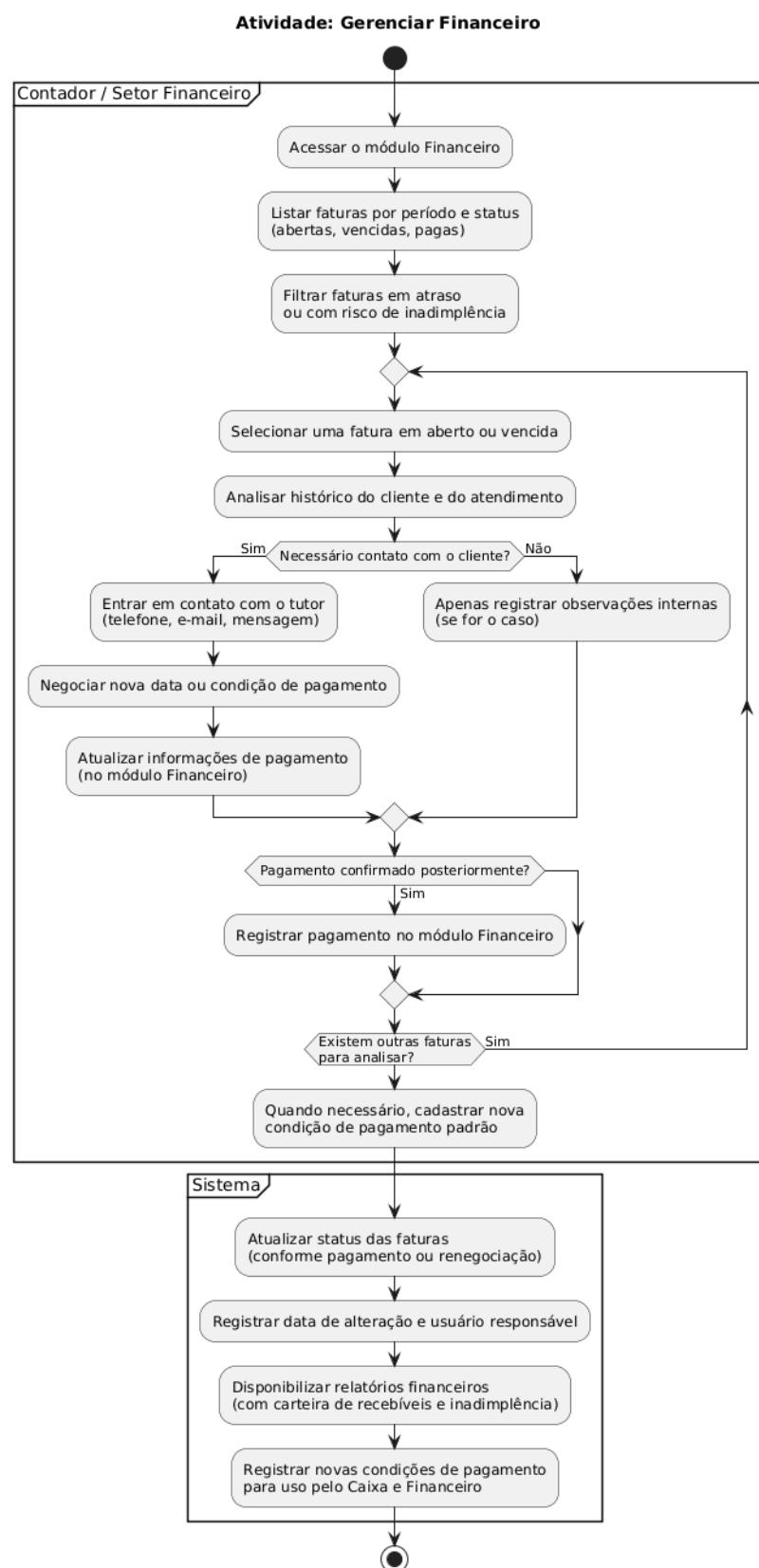


Figura 24. Diagrama de atividades para Gerenciar Financeiro

Gerenciar Cadastros

O diagrama de atividades “Gerenciar Cadastros” representa, de forma genérica, o funcionamento dos principais cadastros do sistema Auravet: tutores, pets, produtos, serviços, condições de pagamento, usuários e funções. O processo inicia quando o usuário autorizado (assistente administrativo, auxiliar, contador ou administrador) acessa o módulo de cadastros desejado e visualiza a lista de registros existentes.

A partir daí, o usuário pode optar por criar um novo registro ou editar um cadastro já existente. No caso de criação, são preenchidos os dados obrigatórios e opcionais conforme o tipo de entidade. Para cadastros de usuários, é necessário vincular o registro a uma função, que define seu perfil de acesso. Para funções, o sistema permite indicar quais módulos podem ser acessados por aquele papel. Nos cadastros de produtos e serviços, são informados preços e o status de ativação, refletindo se podem ser utilizados em atendimentos e faturas. Na edição, o usuário ajusta informações e, quando necessário, marca registros como inativos em vez de excluí-los definitivamente.

O sistema valida campos obrigatórios, formatos e unicidade de informações (como e-mail, CPF ou nomes únicos), graua os dados de forma consistente e disponibiliza esses cadastros para uso integrado pelos módulos de agenda, atendimento e financeiro.

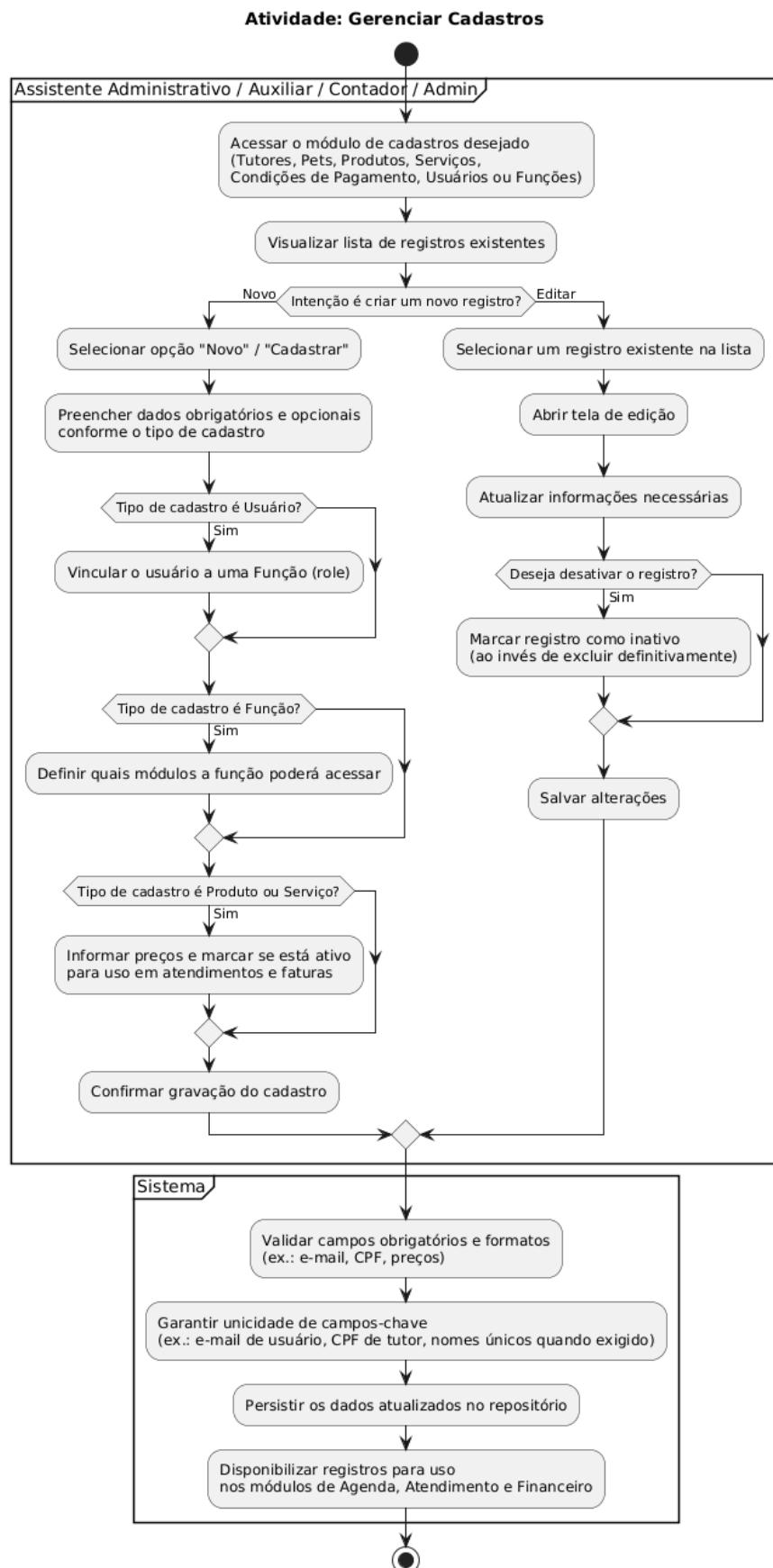


Figura 25. Diagrama de atividades para gerenciar cadastros

2.3.4. Diagramas de sequência

Nesta subseção são apresentados os diagramas de sequência do sistema Auravet, com o objetivo de detalhar a interação temporal entre os atores, a interface de usuário, os serviços de negócio e o repositório de dados. Diferentemente dos diagramas de atividades, que enfatizam o fluxo de ações, os diagramas de sequência destacam a ordem das mensagens trocadas e a colaboração entre os componentes da aplicação em cada cenário.

Foram modelados cinco cenários principais: criação de agendamentos, registro de atendimentos com geração automática de fatura, registro de pagamento no caixa, acompanhamento financeiro de faturas em atraso e gerenciamento genérico de cadastros. Esses diagramas fornecem uma visão clara de como o Auravet orquestra chamadas entre camadas (tela, serviços e banco de dados), garantindo a rastreabilidade das operações mais importantes da clínica.

Criar agendamento

O diagrama de sequência “Criar Agendamento” descreve a interação entre o auxiliar administrativo, a tela de agendamentos (Agenda Inteligente), o serviço de agendamentos e o banco de dados. Inicialmente, o auxiliar acessa a tela de agendamentos, que consulta o serviço de agendamentos para listar horários disponíveis conforme a data e o profissional selecionado. Em seguida, o sistema busca os dados do tutor e do pet já cadastrados e exibe essas informações na interface.

Após definir data, horário, médico, assistente e tipo de atendimento, o auxiliar confirma a criação do agendamento. A tela envia os dados ao serviço de agendamentos, que grava o novo registro no banco de dados com status AGENDADO. Por fim, o serviço retorna os dados do agendamento criado e a interface apresenta uma mensagem de sucesso ao usuário, confirmando a operação.

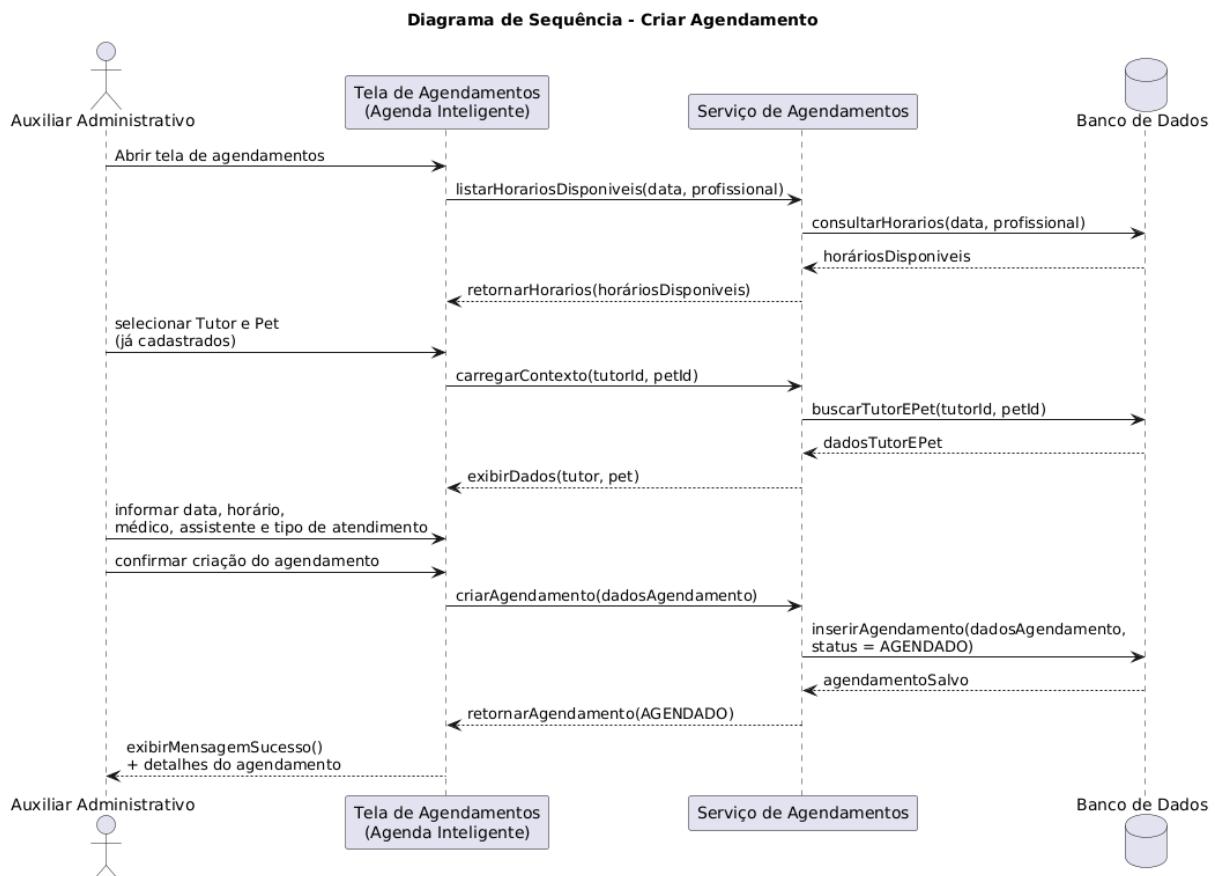


Figura 26. Diagrama de sequência para criar agendamento.

Registrar atendimento e gerar fatura

O diagrama de sequência “Registrar Atendimento e Gerar Fatura” apresenta o fluxo completo do atendimento clínico, desde a seleção do agendamento até a geração automática da fatura. O profissional (médico ou enfermeiro) acessa a Agenda Inteligente, visualiza os agendamentos do dia e seleciona um deles para iniciar o atendimento. A partir dessa ação, o serviço de atendimentos cria um registro de atendimento vinculado ao agendamento e abre a tela específica para lançamento das informações clínicas.

Durante o atendimento, o profissional registra dados clínicos, adiciona serviços realizados e produtos utilizados e inclui notas de prontuário, que são persistidos pelo serviço de atendimentos no banco de dados. Ao concluir o atendimento, o sistema atualiza o status do atendimento e do agendamento para CONCLUÍDO. Em seguida, o serviço de atendimentos aciona o serviço de faturamento, que calcula o valor total com base em serviços e produtos e cria uma fatura vinculada ao tutor e ao

atendimento. Por fim, a tela de atendimento exibe o resumo do atendimento e indica que uma fatura foi gerada automaticamente no módulo de caixa.

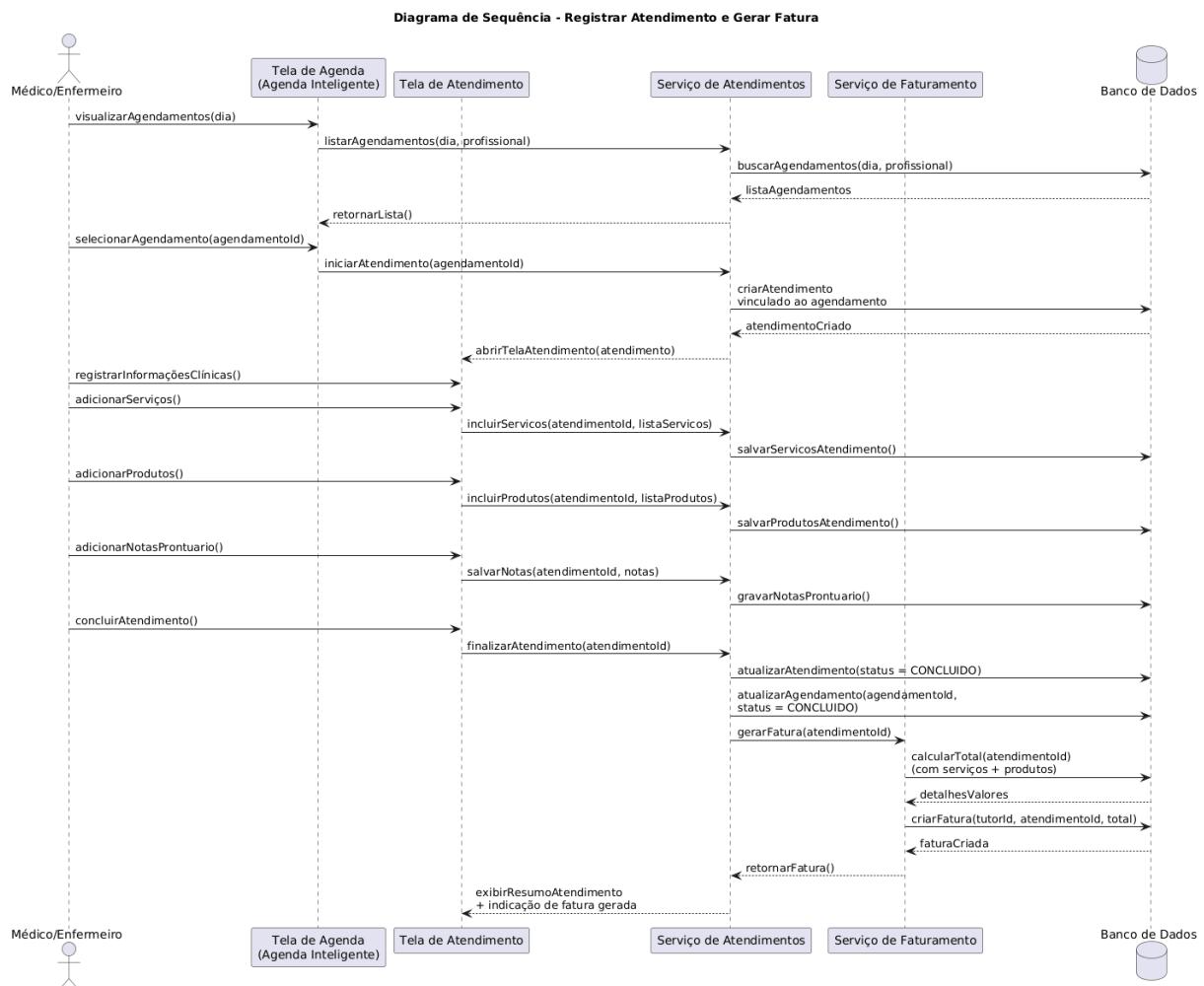


Figura 27. Diagrama de sequência para registrar atendimento e gerar fatura.

Registrar pagamento da fatura

O diagrama de sequência “Registrar Pagamento da Fatura (Caixa)” descreve as interações realizadas pelo assistente ou auxiliar administrativo no momento da cobrança ao cliente. O usuário acessa a tela de caixa, solicita a lista de faturas em aberto e seleciona a fatura correspondente ao atendimento recém-executado. A tela consulta o serviço financeiro, que obtém no banco de dados os detalhes da fatura e seus itens, apresentando-os ao operador.

Em seguida, o colaborador alinha com o cliente a forma e a condição de pagamento, podendo, se necessário, adicionar produtos extras diretamente na

fatura, como vendas de última hora. Esses ajustes são enviados ao serviço financeiro, que atualiza os itens e o valor total. Após a definição final, o assistente informa os dados do pagamento e açãona a função de registrar pagamento, que altera o status da fatura para PAGA e grava a data e demais informações financeiras. Por fim, o sistema açãona o componente de geração de PDF, que produz o documento da fatura/recibo para impressão ou envio ao cliente, concluindo o fluxo no caixa.

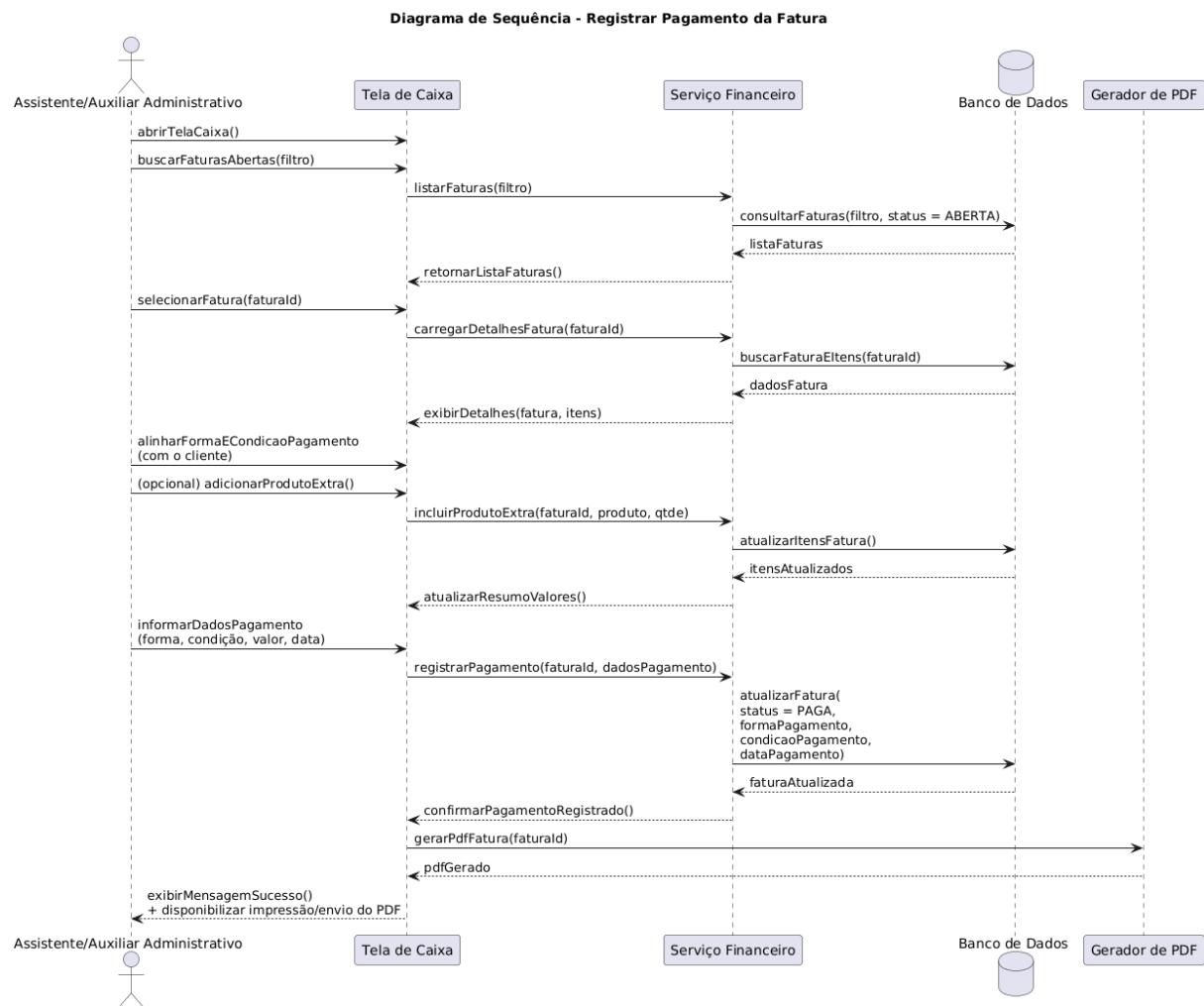


Figura 28. Diagrama de sequência para registrar pagamento da fatura.

Gerenciar Financeiro

O diagrama de sequência “Gerenciar Financeiro (Cobrança)” modela o fluxo de trabalho do contador ou do setor financeiro no acompanhamento de faturas em atraso. Inicialmente, o usuário acessa a tela Financeiro e aplica filtros por período e

status, solicitando a lista de faturas abertas ou vencidas. A tela consulta o serviço financeiro, que recupera do banco de dados as faturas correspondentes e as apresenta para análise.

Em seguida, o contador seleciona uma fatura específica para visualizar seus detalhes e o histórico de ações anteriores. Caso seja necessário realizar uma ação de cobrança (por exemplo, contato telefônico ou por e-mail), essa interação é registrada no sistema, que grava o evento de cobrança associado à fatura. O diagrama também contempla a possibilidade de renegociação da condição de pagamento, na qual novos prazos e condições são salvos e a fatura é atualizada. Quando o pagamento é efetuado após a negociação, o usuário registra a quitação, e o serviço financeiro altera o status da fatura para paga, atualizando a data de pagamento. Esse cenário evidencia o papel do módulo financeiro no controle da carteira de recebíveis e na formalização das ações de cobrança.

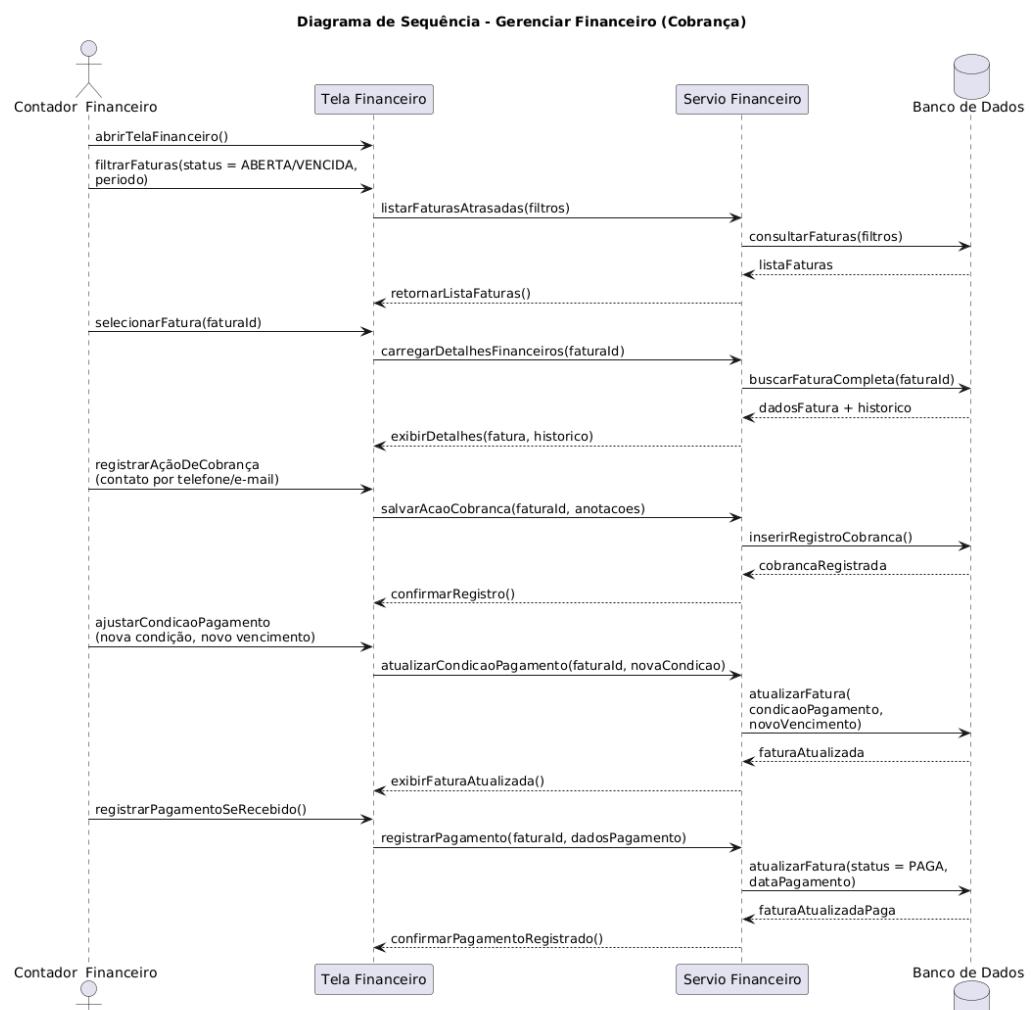


Figura 29. Diagrama de sequência para gerenciar financeiro.

Gerenciar Cadastro

O diagrama de sequência “Gerenciar Cadastros (Genérico)” representa, de forma abstrata, o comportamento comum aos principais cadastros do sistema Auravet, tais como tutores, pets, produtos, serviços, condições de pagamento, usuários e funções. O processo inicia quando o usuário autorizado (assistente administrativo, contador ou administrador) abre a tela de cadastros para um determinado tipo de entidade. A interface solicita ao serviço de cadastros a lista de registros existentes, que é recuperada do banco de dados e exibida para consulta.

A partir dessa visão, o usuário escolhe entre criar um novo cadastro ou editar um registro existente. No cenário de criação, a tela apresenta um formulário em branco, o usuário preenche os dados necessários e confirma a operação. A interface então envia os dados ao serviço de cadastros, que insere o novo registro no banco de dados e retorna a confirmação, atualizando a lista na tela. No cenário de edição, o usuário seleciona um registro, a tela carrega seus dados atuais para edição e, após as alterações, envia as informações atualizadas ao serviço, que grava as mudanças no repositório. Em ambos os casos, o sistema exibe mensagens de sucesso e mantém a consistência dos cadastros utilizados pelos demais módulos do sistema.

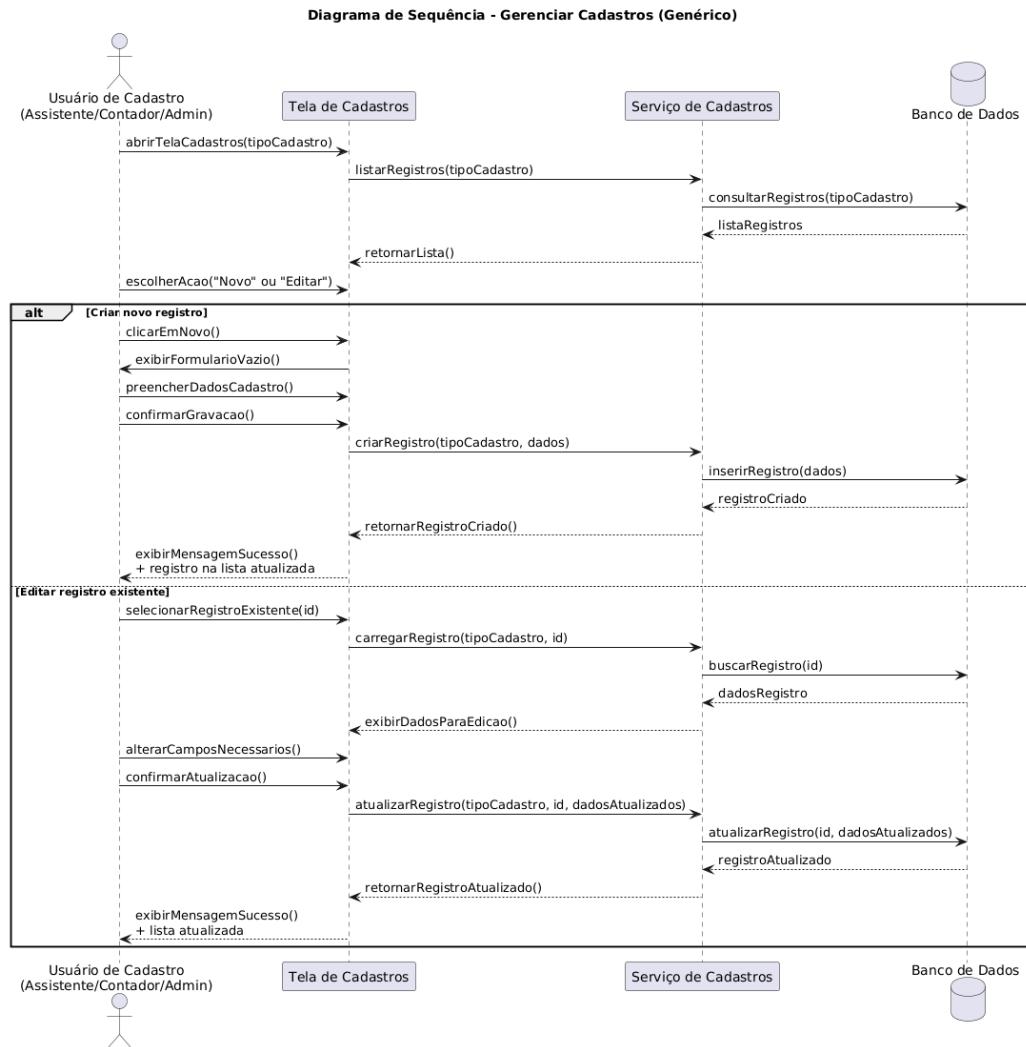


Figura 30. Diagrama de sequências para Gerenciar Cadastros.

2.4. Tecnologias utilizadas

O desenvolvimento do Sistema Auravet foi fundamentado em um conjunto de tecnologias modernas que favorecem desempenho, organização, escalabilidade e experiência de uso.

A escolha de cada ferramenta foi guiada tanto por boas práticas da engenharia de software quanto pela necessidade de criar uma solução leve e agradável, capaz de representar o posicionamento digital-first da clínica.

Nesta seção, são apresentadas as tecnologias aplicadas no frontend, backend, banco de dados e infraestrutura, além das justificativas que sustentam cada decisão técnica adotada.

2.4.1. Frontend

A camada de apresentação da Auravet foi construída utilizando React 18, uma das bibliotecas mais consolidadas para criação de interfaces dinâmicas. A escolha por React permitiu a construção de uma SPA (Single Page Application) fluida e reativa, capaz de lidar com navegação interna eficiente e atualizações de estado em tempo real.

O ambiente de desenvolvimento foi estruturado com Vite 5, que oferece tempos de build reduzidos, recarregamento instantâneo e uma experiência de desenvolvimento mais ágil. A estilização foi realizada com TailwindCSS 3, que possibilita criar interfaces modernas, consistentes e totalmente alinhadas ao design acolhedor da Auravet, garantindo rapidez na prototipagem e excelente controle visual. Complementando essa base, foram utilizados React Router, para navegação entre páginas, React Query, para gerenciamento de estados assíncronos e cache de dados, e React Hook Form, responsável por simplificar formulários e validações front-end. Bibliotecas como Headless UI, Heroicons e Sonner oferecem componentes acessíveis, ícones e notificações que reforçam a experiência refinada da interface.

2.4.2. Backend

A API da Auravet foi desenvolvida com Node.js 20 utilizando o framework Express, que é reconhecido pela leveza, simplicidade e flexibilidade. Essa combinação permite organizar rotas, middlewares e regras de negócio em uma estrutura clara e de fácil manutenção. A validação de dados é realizada com Zod, garantindo que toda entrada enviada para o backend seja verificada, protegendo o sistema de inconsistências e erros silenciosos.

A camada de acesso ao banco de dados utiliza o Prisma ORM, que oferece tipagem automática, migrações estruturadas, queries eficientes e um ecossistema integrado com ferramentas modernas. O Prisma também facilita a modelagem e evolução do schema, mantendo o sistema estável e consistente ao longo das iterações. A documentação das rotas foi implementada com Swagger UI, gerando uma interface acessível para consulta, testes e entendimento da API, o que reforça a transparência e organização do backend.

2.4.3. Banco de Dados

O banco utilizado é o PostgreSQL 16, reconhecido por sua estabilidade, performance e robustez em aplicações modernas. Por ser um banco relacional completo, o PostgreSQL atende perfeitamente às necessidades do sistema, que envolve relacionamentos entre tutores, pets, atendimentos, produtos, serviços, usuários e permissões.

Além disso, o uso do Prisma como camada intermediária reforça a segurança do acesso e garante versionamento e migrações consistentes, evitando divergências entre ambientes distintos.

2.4.4. Infraestrutura

A infraestrutura do Sistema Auravet foi projetada para funcionar tanto em ambiente local quanto em containers. A pasta `infra/docker` reúne tudo o que é necessário para subir o sistema com Docker e Docker Compose, incluindo containers para a API, o banco de dados PostgreSQL e o build estático do frontend. Essa abordagem garante previsibilidade, padronização e facilita a replicação do ambiente de desenvolvimento ou execução em diferentes máquinas.

O sistema também utiliza npm workspaces, que organizam o monorepo e permitem gerenciar dependências e scripts de forma unificada, mantendo API e frontend alinhados. Ferramentas como Husky e lint-staged foram integradas ao fluxo de desenvolvimento para garantir formatação, lint e tipagem antes dos commits, preservando a saúde do código.

2.4.5. Justificativa das Tecnologias

A combinação de React, Node.js, Prisma e PostgreSQL proporciona uma base moderna, estável e familiar para desenvolvedores, além de permitir a criação de um sistema modular e escalável. A escolha por Vite e Tailwind reforça a intenção de oferecer uma interface rápida, visualmente consistente e alinhada à estética da Auravet. Da mesma forma, o uso de Docker permite que o projeto seja facilmente distribuído e executado em qualquer ambiente sem necessidade de configurações complexas.

Todas as tecnologias escolhidas refletem o compromisso com simplicidade, qualidade técnica e experiência agradável, valores que conversam diretamente com a proposta da Auravet de entregar uma plataforma leve, acolhedora e eficiente.

2.5. Branding e diretrizes visuais

A identidade visual do Sistema Auravet foi concebida para refletir o posicionamento emocional, estético e conceitual da clínica. Como a Auravet nasceu com a proposta de unir tecnologia, empatia e cuidado humanizado, o design cumpre um papel essencial para transmitir essa sensibilidade diretamente na experiência do usuário. Cada elemento visual – desde as cores até a tipografia – foi selecionado para reforçar essa atmosfera de acolhimento, leveza e profissionalismo, criando um ambiente digital que dialoga com a essência da marca.

A escolha cromática, tipográfica e iconográfica está presente em todas as áreas do sistema, garantindo unidade visual entre telas, componentes, mensagens e interações. Essa coesão não é apenas estética: ela melhora a usabilidade, orienta a navegação e fortalece a identidade da Auravet enquanto produto digital. A seguir, são descritos os principais aspectos que moldam o estilo visual da plataforma.

2.5.1. Paleta de cores

A paleta da Auravet foi construída para transmitir serenidade, confiança e naturalidade. O verde-sálvia (A7C7AO) expressa calma e cuidado; o azul-claro (B3D4EO) sugere confiança e clareza; o verde-escuro (3D6655) complementa com profundidade; o branco gelo (F8FAF9) traz leveza; e o grafite profundo (OF172A) trabalha como cor de contraste e legibilidade. Esses tons foram aplicados de forma equilibrada ao longo de todo o sistema, reforçando seu caráter acolhedor e moderno.



Figura 31. Paleta de cores Auravet

2.5.2. Tipografia

A tipografia desempenha papel crucial na percepção da marca. O sistema utiliza Montserrat como fonte principal para títulos, conferindo firmeza, modernidade e boa presença visual, enquanto Nunito Sans é aplicada aos textos corridos, trazendo suavidade, acessibilidade e excelente leitura em diferentes tamanhos. A combinação das duas mantém uma hierarquia clara e contribui para a sensação de profissionalismo e cuidado presente em toda a interface.

Figura 32. Exemplo de aplicação da Tipografia na tela de Tutores

2.5.3. Logo

O logotipo oficial da Auravet sintetiza de forma simples e carismática a essência da clínica. Ele apresenta um coração estilizado em tons de verde, mas com uma personalidade lúdica: duas formas arredondadas representam olhos, enquanto um sorriso discreto cria uma expressão acolhedora e amigável. Essa combinação reforça a ideia de que o cuidado veterinário envolve não apenas técnica, mas também vínculo e afeto.

A escolha por um símbolo minimalista – composto por poucos elementos geométricos e uma paleta suave – dialoga diretamente com a identidade da Auravet, que aposta em leveza, empatia e naturalidade como pilares do seu design. O coração central representa proteção e carinho; o verde remete à natureza e ao bem-estar; e a expressão soridente transmite confiança, acolhimento e humanidade.



Figura 33. Logo da Auravet

Esse ícone foi pensado para funcionar bem em telas modernas, mantendo legibilidade tanto em tamanhos reduzidos (como favicons, avatares e ícones mobile) quanto em aplicações maiores, como áreas de destaque do painel e materiais visuais complementares. A estética redonda e amigável reforça a sensação de que a plataforma é acessível e projetada para acolher não apenas os pets, mas também os profissionais que fazem parte da rotina da clínica.

2.5.4. Manual de UI

O design system da Auravet foi organizado para garantir consistência entre todos os elementos da interface. Componentes como botões, cards, tabelas, inputs, modais e formulários seguem padrões rígidos de espaçamento, cor e tipografia, sempre buscando equilíbrio entre estética e funcionalidade.

Animais

Cada pet carrega uma história. Visualize tutores, espécie e histórico de atendimentos em um só lugar.

Pets acompanhados

Toque em um pet para abrir o histórico completo.

Exportar lista

Buscar por pet Nome do pet **Tutor** Nome do tutor **Espécie** Todas

Ônix
Gato • Tutor(a): Nair Adriana Caldeira
BENGAL

Eco
Gato • Tutor(a): Stella Larissa Vieira
SIAMÉS

Zéfiro
Cachorro • Tutor(a): Vicente Elias Miguel da Cunha
BORDER COLLIE

Corvo
Gato • Tutor(a): Calebe Julio Rodrigues
BOMBAY

Histórico do pet
Atendimentos realizados com datas e valores.

Ônix
Gato • Tutor(a): Nair Adriana Caldeira
NASCIMENTO: 08/03/2020

Contato do tutor
nair.caldeira@focusdm.com.br
14997774000
CPF: 199.513.588-70
Rua Carolina Moraes Almeida, 545 • Senador Salgado Filho • Marília - SP • CEP 17502-290

Registrar Atendimento

Exportar atendimentos

Abrir histórico completo

Início dd/mm/aaaa **Fim** dd/mm/aaaa **Status** Todos

CONSULTA
22/11/2025 • R\$ 220.00 **Editar**
Serviço Avaliação pré-cirúrgica realizado durante o agendamento seed.

Figura 34. Exemplo de UI na tela de Animais

O uso do TailwindCSS permitiu transformar essas diretrizes em classes reutilizáveis, mantendo coesão em toda a plataforma e facilitando futuras expansões. O sistema privilegia clareza, contraste adequado, acessibilidade e interação intuitiva, reforçando o propósito da clínica de proporcionar um ambiente digital simples e agradável de usar.

2.5.5. Tom de voz

O tom de voz adotado no sistema e em toda a comunicação textual é empático, acolhedor e didático. As mensagens – sejam elas instruções, avisos, etiquetas ou validações – foram escritas para tranquilizar o usuário, orientar com clareza e reforçar a abordagem humanizada da Auravet. Em vez de mensagens técnicas e frias, a comunicação busca transmitir proximidade e confiança, sem perder precisão ou objetividade.

Figura 35. Exemplo do tom de voz adotado na tela de atendimento

A integração desses elementos – cor, tipografia, identidade, componentes e linguagem – forma o núcleo estético do Sistema Auravet, garantindo que cada interação reforce a ideia de um cuidado natural, humano e tecnicamente confiável. O branding não é apenas um complemento visual, mas um pilar que permeia toda a experiência e traduz digitalmente o propósito da clínica.

2.6. Segurança do sistema

A segurança do Sistema Auravet foi planejada desde o início como um pilar estrutural, garantindo que cada interação do usuário ocorra dentro de um ambiente protegido, confiável e alinhado às boas práticas de aplicações web modernas.

A implementação combina camadas de proteção no backend, validações rigorosas de entrada, controle granular de permissões e mecanismos de mitigação de ataques comuns.

Dessa forma, o sistema busca não apenas impedir acessos indevidos, mas também preservar a integridade dos dados, evitar exposição de informações sensíveis e reforçar a autenticidade de cada operação realizada na plataforma.

2.6.1. Autenticação

A autenticação na Auravet é totalmente baseada em JWT (JSON Web Tokens), emitidos no momento do login e validados a cada requisição protegida. Para reforçar a segurança, o token inclui informações essenciais do usuário autenticado, como o conjunto de módulos que determinam suas permissões, permitindo que o backend execute verificações rápidas e sem ambiguidade. O algoritmo de assinatura utilizado é o HS256, com chave definida via variável de ambiente (JWT_SECRET) e prazo de expiração configurável (JWT_EXPIRES_IN).

O processo de login é cuidadosamente protegido. Antes de qualquer autenticação, o payload é validado por meio de schemas Zod, assegurando que apenas dados bem formados sejam processados. As tentativas de login passam também por um mecanismo de rate limiting configurável, reduzindo riscos de ataques de força bruta. Durante a verificação das credenciais, as senhas são comparadas utilizando funções que evitam ataques de timing.

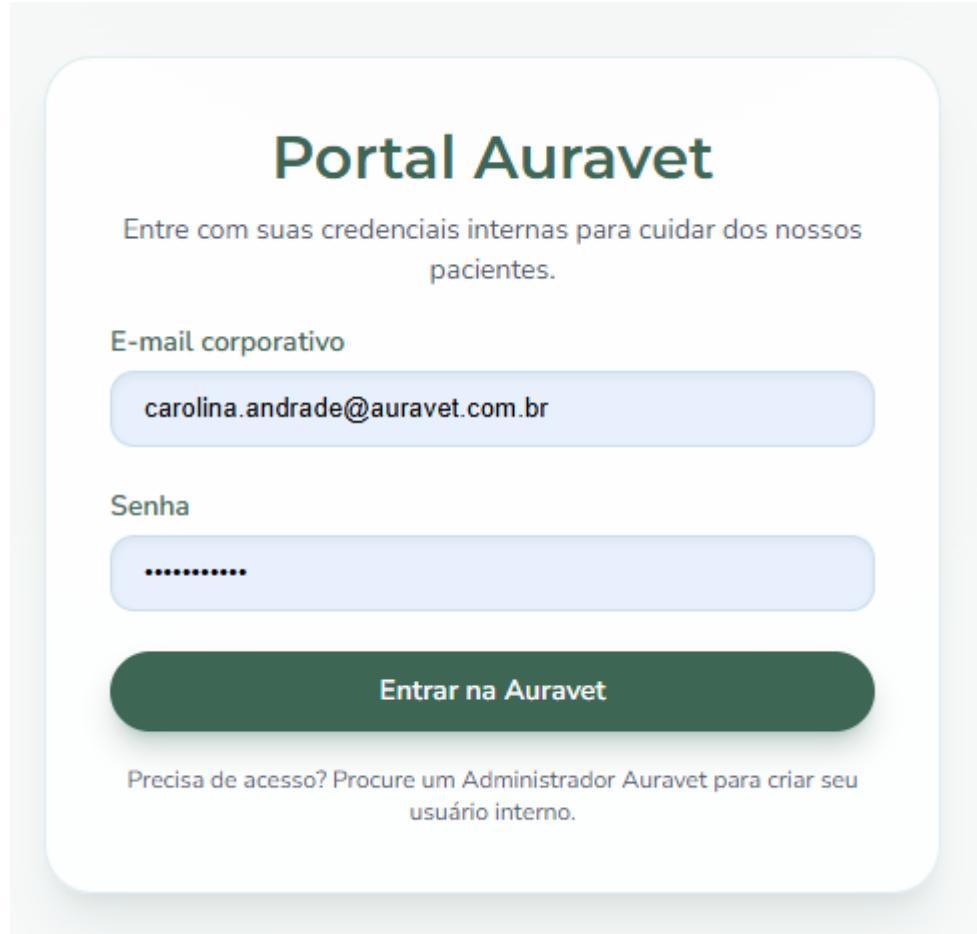


Figura 36. Tela de login da Auravet

Quando o token é enviado ao servidor, o middleware de autenticação extrai, valida e interpreta o Bearer Token, consultando o banco via Prisma para confirmar se o usuário existe, está ativo e possui permissões devidamente associadas ao seu papel. Caso qualquer etapa falhe, a requisição é encerrada imediatamente com um erro seguro e padronizado.

```
GET /auth/me 200 16.012 ms - 761
prisma:query SELECT 1
prisma:query SELECT "public"."User"."id", "public"."User"."nome", "public"."User"."email", "public"."User"."passwordHash", "public"."User"."roleId", "public"
"User"."createdAt", "public"."User"."updatedAt" FROM "public"."User" WHERE ("public"."User"."email" = $1 AND 1=1) LIMIT $2 OFFSET $3
prisma:query SELECT "public"."Role"."id", "public"."Role"."name", "public"."Role"."slug", "public"."Role"."description", "public"."Role"."isActive", "public"
"Role" WHERE "public"."Role"."id" IN ($1) OFFSET $2
prisma:query SELECT "public"."RoleModuleAccess"."roleId", "public"."RoleModuleAccess"."isEnabled" FROM "public"."RoleModuleAccess" WHERE ("public"."RoleModuleAccess"."isEnabled" = $1 AND ("j1"."isActive" = $2 AND ("j1"."id" IS NOT N
OFFSET $4
prisma:query SELECT "public"."Module"."id", "public"."Module"."name", "public"."Module"."slug", "public"."Module"."description", "public"."Module"."isActive"
"public"."Module"."updatedat" FROM "public"."Module" WHERE "public"."Module"."id" IN ($1,$2,$3,$4,$5,$6) OFFSET $7
prisma:query SELECT "public"."CollaboratorProfile"."id", "public"."CollaboratorProfile"."userId", "public"."CollaboratorProfile"."especialidade", "public"."C
"public"."CollaboratorProfile"."turnos", "public"."CollaboratorProfile"."bio", "public"."CollaboratorProfile"."createdAt", "public"."CollaboratorProfile"."up
"public"."CollaboratorProfile"."userId" IN ($1) OFFSET $2
POST /auth/login 401 16.405 ms - 208
Error: Credenciais inválidas.
at /app/dist/routes/auth.js:46:15
```

Figura 37. Print do Docker quando as credenciais estão incorretas

2.6.2. Autorização

A autorização segue um modelo baseado em módulos, permitindo granularidade fina para controlar quais partes da aplicação podem ser acessadas por cada função. Cada colaborador possui um papel (como Administrador, Médico, Enfermeiro ou Auxiliar Administrativo), e cada papel é composto por um conjunto de módulos ativos que habilitam determinadas ações dentro do sistema.

Essa lógica é aplicada no backend por meio de middlewares como `requirePermission` e `requireAnyPermission`, que validam em tempo real se o usuário autenticado possui permissão explícita para acessar a rota. Se a permissão estiver ausente, o backend retorna respostas claras – 401 para ausência de autenticação e 403 para falta de autorização – mantendo a integridade da aplicação e impedindo acessos indevidos.

The screenshot shows a web application interface for managing functions and modules. At the top, there's a header with the title "Funções e módulos" and a button labeled "Nova função". Below the header, a sub-header says "Funções cadastradas" with a note: "Organize a equipe conforme responsabilidades e mantenha os módulos sob controle." A "Nova função" button is also present here. The main content area displays two roles: "Administrador" and "Assistente Administrativo". Each role has a list of assigned modules and specific function buttons. For "Administrador", the modules are "Ajustar módulos", "Desativar", and "Remover". The assigned functions include "Administrar usuários", "Caixa", "Gerenciar animais", "Gerenciar produtos", "Gerenciar serviços", "Gerenciar tutores", "Visualizar animais", "Visualizar produtos", "Visualizar serviços", and "Visualizar tutores". For "Assistente Administrativo", the modules are "Ajustar módulos", "Desativar", and "Remover". The assigned functions include "Caixa", "Gerenciar serviços", "Visualizar animais", "Visualizar produtos", "Visualizar serviços", and "Visualizar tutores".

Figura 38. Tela Auravet para cadastro de funções

Essa estrutura torna a plataforma flexível e alinhada ao fluxo real de uma clínica veterinária, onde diferentes profissionais operam funções específicas e devem acessar apenas as áreas compatíveis com sua responsabilidade.

2.6.3. Proteções e Boas Práticas

Além dos mecanismos principais de autenticação e autorização, o sistema incorpora uma série de proteções complementares que reforçam a robustez do backend:

A primeira delas é o uso do helmet, responsável por configurar cabeçalhos HTTP seguros automaticamente. Com ele, são ativadas proteções contra-ataques como XSS, Clickjacking e MIME sniffing, reduzindo a superfície de vulnerabilidade típica de APIs públicas. Complementando essa segurança, o middleware CORS é configurado de forma estritamente controlada por meio de variáveis de ambiente, limitando quais origens podem se comunicar com a API e reforçando a proteção contra requisições não autorizadas.

Outro ponto crítico é a validação das variáveis de ambiente. Antes mesmo de iniciar o servidor, o sistema aplica um schema Zod para garantir que configurações essenciais como DATABASE_URL, JWT_SECRET e parâmetros de autenticação estejam presentes e válidas. Caso alguma delas esteja ausente ou incorreta, a aplicação sequer inicia, evitando estados inseguros ou ambiguidades.

O armazenamento de senhas segue um padrão rigoroso: todas são processadas usando scrypt com salt gerado dinamicamente e custo configurável por variável de ambiente (PASSWORD_SALT_ROUNDS). O sistema define valores mínimos e máximos, garantindo que mesmos parâmetros configurados incorretamente não comprometam a segurança. As comparações de senha utilizam técnicas livres de tempo variável, minimizando risco de ataques baseados em medição de tempo.

Além disso, operações sensíveis – como login e registro – possuem rate limiting dedicado, impedindo tentativas excessivas e dificultando ataques automatizados. Já o tratamento de payloads é conduzido por schemas Zod em praticamente todas as rotas, assegurando que nenhuma informação malformada ou potencialmente perigosa chegue às regras de negócio ou ao banco de dados.

2.7. Testes automatizados

A estratégia de testes da Auravet foi planejada para oferecer confiança no comportamento do sistema e prevenir regressões nos módulos mais sensíveis,

como autenticação, autorização, agendamentos e controle de permissões. Embora não tenha sido o foco central do projeto, a aplicação implementa um conjunto sólido de testes automatizados que ajudam a validar as regras de negócio e garantem a estabilidade do monorepo à medida que novas funcionalidades são adicionadas.

No backend, os testes foram construídos utilizando o runner nativo do Node.js (`node:test`), permitindo executar suítes simples e diretas sem necessidade de bibliotecas externas. O ponto mais importante da abordagem é que a API é levantada durante os testes em uma porta provisória (`app.listen(0)`), enquanto as operações de banco de dados são executadas sobre um Prisma em memória, criado por meio da função `createInMemoryPrisma`. Essa técnica elimina dependências externas e cria um ambiente isolado, rápido e reproduzível, ideal para validar rotas e integrações leves sem envolver um PostgreSQL real.

A API conta com diferentes suítes que cobrem áreas críticas. O módulo de autenticação possui um conjunto completo de testes que verifica fluxos de login – incluindo credenciais válidas, inválidas e tentativas com payload incorreto – além do registro de colaboradores, checagem de papéis, herança de permissões e compatibilidade entre diferentes custos de hashing de senha. Esse conjunto garante que todo o processo de acesso à API permaneça previsível e seguro, mesmo com ajustes futuros no esquema de roles ou nas políticas de autenticação.

O controle de acesso também é validado em outra suíte dedicada ao módulo de tutores. Ela verifica, por exemplo, que perfis como contador ou caixa conseguem acessar apenas a versão básica dos dados, sendo corretamente bloqueados ao tentar visualizar detalhes mais sensíveis. Esses testes reforçam a integridade do sistema de permissões baseado em módulos, que é um dos pilares da segurança da Auravet.

Outra área sensível analisada pelos testes é o fluxo de agendamentos e atendimentos. A suíte dedicada valida o comportamento das rotas de atualização e conclusão de atendimentos, garantindo que a alteração de status reflita corretamente no faturamento relacionado e que o schema de validação (`appointmentUpdateSchema`) trate adequadamente casos como a desconexão de assistentes – mantendo `assistantId` como nulo quando apropriado. O Prisma é

totalmente mockado neste contexto, permitindo simular diferentes cenários de atualização sem manipular dados reais.

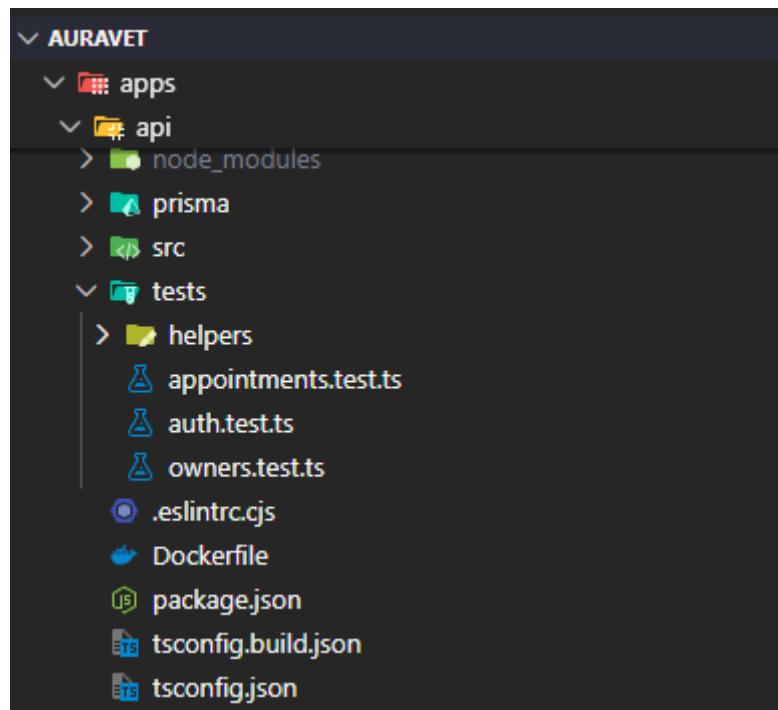


Figura 39. Diretório de teste do backend

No frontend, há um teste adicional que atua como um “smoke test” voltado especificamente ao controle de permissões. O arquivo `RequireModules.test.ts` valida a função `hasAllRequiredModules`, garantindo que o componente responsável por bloquear ou permitir acesso às páginas funcione corretamente. Esse teste, embora simples, serve como uma garantia importante para evitar regressões no fluxo de navegação dos colaboradores, reforçando a coerência entre as permissões aplicadas no backend e a experiência exibida ao usuário.

```

ocaio@opcdocao MINGW64 /c/git/auravet (main)
$ npm run test --workspace apps/api -- tests/auth.test.ts

> auravet-api@1.0.0 test
> tsx --test tests/**/*.test.ts tests/auth.test.ts

▶ appointmentUpdateSchema
  ✓ retains null assistantId values to allow disconnection (102.8558ms)
✓ appointmentUpdateSchema (1297.6102ms)
POST /auth/login 200 84.337 ms - 977
PATCH /appointments/cd7a52d91c196442e9514cea5/complete 200 15.218 ms - 1933
▶ PATCH /appointments/:id/complete
  ✓ updates invoices when completing an appointment with an existing service invoice (391.6292ms)
✓ PATCH /appointments/:id/complete (392.4432ms)
POST /auth/login 200 61.969 ms - 977
▶ Authentication flows
  ✓ allows a user to login with valid credentials (297.2625ms)
POST /auth/login 401 71.574 ms - 246
  ✓ rejects login with invalid credentials (150.242ms)
Error: Credenciais inválidas.
  at <anonymous> (C:/git/auravet/apps/api/src/routes/auth.ts:55:13)
POST /auth/login 200 54.988 ms - 977
POST /auth/register 201 47.867 ms - 473
  ✓ allows administrators to register new collaborators (195.3238ms)
POST /auth/login 200 44.825 ms - 977
POST /auth/register 201 53.377 ms - 478
  ✓ allows administrators to register new collaborators using role slug identifiers (167.9886ms)
POST /auth/login 200 44.881 ms - 856
POST /auth/register 403 3.439 ms - 697
  ✓ blocks non administrators from creating users (148.9805ms)
Error: Você não tem permissão para acessar este recurso.
  at <anonymous> (C:/git/auravet/apps/api/src/middlewares/require-permission.ts:18:13)
  at Layer.handle [as handle_request] (C:/git/auravet/node_modules/express/lib/router/layer.js:95:5)
  at next (C:/git/auravet/node_modules/express/lib/router/route.js:149:13)
  at <anonymous> (C:/git/auravet/apps/api/src/middlewares/authenticate.ts:53:5)
  at process.processTicksAndRejections (node:internal/process/task_queues:105:5)
POST /auth/login 200 88.996 ms - 977
  ✓ allows a user to login when the password hash uses a different cost (232.0531ms)
POST /auth/login 200 77.983 ms - 977
  ✓ allows a user to login with a legacy password hash after cost changes (161.4246ms)
✓ Authentication flows (2719.644ms)
POST /auth/login 200 45.533 ms - 977
PATCH /roles/ADMINISTRADOR/modules 200 16.231 ms - 2701

```

Figura 40. Print da execução do auth.test.ts

Em conjunto, esses testes formam uma base confiável para apoiar o desenvolvimento contínuo do sistema. Eles priorizam os pontos estruturais do domínio – autenticação, autorização, acessos restritos e atualização de dados críticos – contribuindo para que a Auravet permaneça estável e previsível mesmo com evoluções constantes em suas funcionalidades.

2.8. Padrões de código e organização

O código da Auravet foi organizado para ser fácil de ler, manter e evoluir. Todo o projeto está estruturado em um monorepo com dois workspaces principais (apps/api e apps/web), ambos escritos em TypeScript com configuração estrita. Essa abordagem garante tipagem forte em toda a solução, reduzindo erros em tempo de execução e facilitando o refino das regras de negócio e da interface.

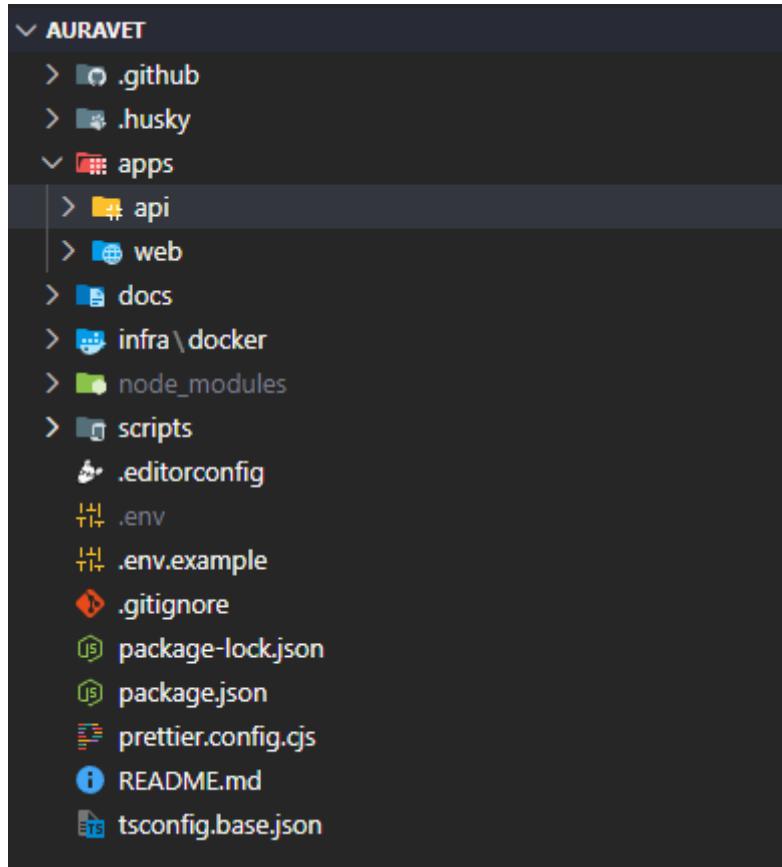


Figura 41. Estrutura do Monorepo

Os scripts de build, lint, formatação e verificação de tipos ficam centralizados na raiz do projeto, o que permite executar checagens globais com um único comando. Isso ajuda a manter a API e o frontend alinhados, mesmo evoluindo em ritmos diferentes.

2.8.1. Convenções

O projeto utiliza uma configuração de Prettier compartilhada para unificar o estilo de escrita do código. A formatação segue largura de linha de 100 caracteres, uso de aspas simples, vírgula final em estruturas como objetos e arrays e ponto e vírgula obrigatório. Esses detalhes podem parecer pequenos, mas eliminam discussões de estilo e deixam o foco do time voltado para a lógica do sistema em vez de formatação.

Para a qualidade estática, são utilizadas configurações de ESLint especializadas por contexto. Na API, o lint é voltado para TypeScript e backend: há regras de ordenação de imports, checagem de promessas não tratadas e presets específicos

para módulos TypeScript/Import. No frontend, o ESLint é ajustado para React, com presets para React, Hooks e Acessibilidade (A11y), além do suporte a TypeScript. Regras desnecessárias para o contexto atual, como react/react-in-jsx-scope, foram desabilitadas por conta do uso de JSX moderno, evitando ruídos desnecessários no fluxo de desenvolvimento.

```

1 {           Caio Murillo, mês passado | 1 author (Caio Murillo)
2   "name": "auravet",
3   "version": "1.0.0",
4   "private": true,
5   "workspaces": [
6     "apps/web",
7     "apps/api"
8   ],
9   >Depurar
10  "scripts": {
11    "build": "npm run build -ws",
12    "lint": "npm run lint -ws",
13    "format": "npm run format -ws",
14    "typecheck": "npm run typecheck -ws",
15    "dev": "concurrently \"npm run dev --workspace apps/api\" \"npm run dev --workspace apps/web\"",
16    "prepare": "husky install"
17  },
18  "devDependencies": {
19    "concurrently": "^8.2.2",
20    "husky": "^9.0.11",
21    "lint-staged": "^15.2.2",
22    "prettier": "^3.2.5"
23  },
24  "lint-staged": {
25    "*.{ts,tsx,js,jsx,json,css,md)": [
26      "prettier --write"
27    ]
28  }
}

```

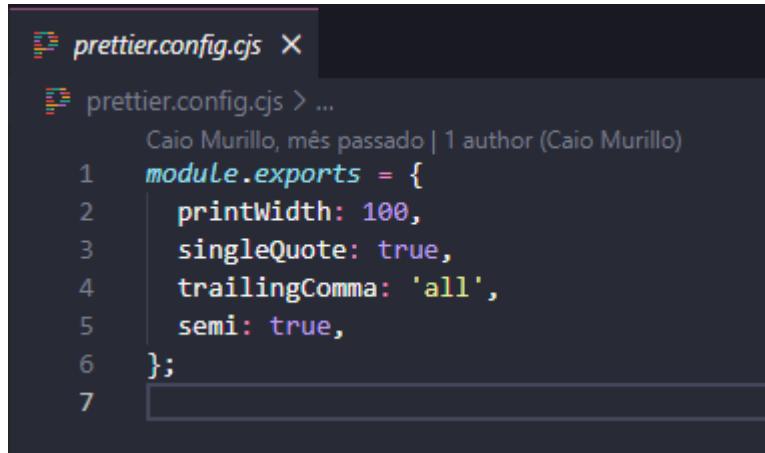
Figura 42. Trecho do package.json da raiz com os scripts de qualidade

Essas convenções formam um “contrato visual” do código: independente de quem escreva, o resultado segue o mesmo padrão, facilitando a leitura em qualquer arquivo do monorepo.

2.8.2. Linting e Formatação

Na raiz do projeto, estão definidos os scripts npm run lint, npm run format e npm run typecheck, configurados para serem executados sobre todos os workspaces. Isso permite rodar uma bateria completa de verificações antes de builds ou testes, garantindo que a base esteja consistente e dentro dos padrões estabelecidos.

Cada aplicação também expõe seus próprios comandos de lint e formatação, permitindo que o desenvolvedor foque apenas na API ou apenas no frontend quando necessário. Essa flexibilidade ajuda no dia a dia: é possível rodar checagens rápidas em um módulo específico ou realizar inspeções globais antes de uma entrega mais importante.



```

P prettier.config.cjs X
P prettier.config.cjs > ...
Caio Murillo, mês passado | 1 author (Caio Murillo)
1 module.exports = {
2   printWidth: 100,
3   singleQuote: true,
4   trailingComma: 'all',
5   semi: true,
6 };
7

```

Figura 43. Print de um arquivo de configuração de lint/format

2.8.3. Hooks de desenvolvimento

Para garantir que todas essas regras sejam respeitadas na prática, o repositório utiliza Husky e lint-staged integrados ao fluxo de versionamento. Um hook de pre-commit executa automaticamente npm run lint e npm run typecheck, bloqueando commits que violem regras de estilo ou apresentem problemas de tipos. Dessa forma, erros mais básicos nem chegam a entrar no histórico do Git.

O lint-staged atua em complemento: ele roda o Prettier apenas nos arquivos modificados em cada commit (*.{ts,tsx,js,jsx,json,css,md}), aplicando formatação automática antes de finalizar o versionamento. O resultado é um repositório com histórico limpo, formatação consistente e menos retrabalho com ajustes manuais de código.

2.9. Guia de Deploy

O repositório da Auravet foi pensado para ser simples de subir em qualquer ambiente de desenvolvimento, oferecendo dois caminhos principais de execução: modo local, utilizando Node.js e PostgreSQL instalados na máquina, e stack

completa via Docker Compose, encapsulando banco, API e frontend em containers. Ambos os fluxos estão documentados no README e compartilham a mesma base de configuração por variáveis de ambiente, destacando valores críticos como DATABASE_URL, JWT_SECRET e VITE_API_URL, além dos comandos de migração e seed do banco.

A presença de scripts automatizados em shell torna o processo de preparação do ambiente muito mais previsível. Em vez de uma sequência longa de comandos manuais, o desenvolvedor executa um script único que cuida de instalar dependências, preparar o banco de dados, rodar o seed e iniciar a aplicação. Isso reduz erros humanos, facilita demonstrações em laboratório e aproxima o projeto de um cenário de deploy mais profissional.

2.9.1. Execução Local

No modo local, a Auravet é executada diretamente sobre o ambiente do desenvolvedor, partindo do pressuposto de que Node.js e PostgreSQL já estão instalados e acessíveis. Para simplificar esse processo, o repositório oferece o script scripts/setup-local.sh, responsável por orquestrar todas as etapas de preparação do monorepo.

Ao ser executado, o script verifica primeiro a existência do arquivo .env na raiz do projeto. Caso ele ainda não exista, é criado automaticamente a partir de .env.example, garantindo que todas as variáveis necessárias tenham um valor inicial. Em seguida, o script instala as dependências do monorepo com npm install, gera o Prisma Client para a API, aplica as migrações ao banco e executa o seed responsável por criar os dados iniciais – incluindo o usuário administrador, se as variáveis correspondentes estiverem configuradas.

Depois que o setup-local.sh termina, o ambiente está pronto para uso. A execução simultânea da API e do frontend é feita por meio do comando único npm run dev, que levanta o servidor Express na porta 4000 e o Vite na porta 5173. Esse fluxo reduz o esforço de configuração manual e permite que qualquer integrante do time comece a testar o sistema rapidamente em sua própria máquina.

2.9.2. Execução com Docker

Para cenários em que se deseja isolar completamente o ambiente, o projeto disponibiliza uma stack baseada em Docker Compose, localizada em infra/docker. Esse modo de execução sobe três serviços principais: um container com o PostgreSQL, um container para a API Node/Express e outro para o frontend gerado com Vite, servido por Nginx. A configuração permite ajustar portas, credenciais de banco e segredos via .env, mantendo a interface de variáveis utilizada no modo local.

O script scripts/setup-docker.sh automatiza todo o fluxo de criação dos containers. Ele começa derrubando e limpando containers e volumes antigos, zerando o estado do banco de dados para garantir um ambiente previsível. Em seguida, executa o build das imagens da API e do frontend, respeitando as configurações de ambiente definidas na raiz. Depois disso, sobe apenas o serviço de banco de dados e aguarda sua disponibilidade.

Com o banco no ar, o script aplica as migrações Prisma por meio do comando prisma migrate deploy rodando dentro do container da API. Na sequência, é executada uma rotina de seed que compila prisma/seed.ts com tsc para uma pasta prisma/dist e, posteriormente, roda o seed.js gerado, populando a base com dados iniciais. Só então a API e o frontend são efetivamente levantados em modo detach, deixando a stack pronta para uso com a API exposta em <http://localhost:4000> e a interface web em <http://localhost:5173>.

Os Dockerfiles foram organizados para refletir boas práticas de build em múltiplas fases. A imagem da API utiliza uma etapa de build que gera o bundle TypeScript, o Prisma Client e os artefatos necessários, copiando apenas o resultado para uma imagem node 20 mais enxuta, o que reduz tamanho e superfície de ataque. Já o frontend gera o build estático do Vite em uma imagem node 20 Alpine e, em seguida, copia os arquivos para uma imagem Nginx que serve a aplicação na porta 80. Em ambos os casos, a variável VITE_API_URL é essencial no momento do build, pois define para qual Endpoints a SPA irá apontar nas chamadas de API.

```
[+] Running 3/3
✓ Container auravet-db  Healthy
✓ Container auravet-api  Started
✓ Container auravet-web  Started
✓ Pronto! Acesse web em http://localhost:5173 e API em http://localhost:4000
```

Figura 44. Conclusão da execução via Docker.

O guia de deploy reforça ainda que, para qualquer ambiente que simule produção, DATABASE_URL e JWT_SECRET não devem usar valores padrão. O próprio código de configuração aborta a inicialização se essas variáveis estiverem vazias, prevenindo deploys inseguros. O Compose permite sobrepor parâmetros de banco e portas conforme necessidade, mantendo a estrutura de scripts para migração e seed, o que facilita a replicação do ambiente em diferentes máquinas ou servidores.

2.10. Referências de API

A API da Auravet segue o padrão REST e é documentada automaticamente utilizando Swagger (OpenAPI 3.0). Toda a documentação técnica pode ser acessada em: <http://localhost:4000/docs>

The screenshot shows the Swagger UI interface for the Auravet API. At the top, it displays the title "Auravet API" with version "1.0.0" and "OAS 3.0". Below the title, there's a brief description: "API oficial da clínica veterinária Auravet. Todos os endpoints retornam respostas JSON e utilizam validação com Zod." It also includes links to "Auravet - Website" and "Send email to Auravet".

The main area is titled "Servers" and shows "http://localhost:4000 - Ambiente local de desenvolvimento". To the right of this is an "Authorize" button with a lock icon.

Under the "default" section, there are three API endpoints listed:

- GET /health**: Checa disponibilidade da API
- POST /auth/login**: Autentica um colaborador interno
- POST /auth/register**: Cadastra um novo colaborador interno

Figura 45. Página da documentação do Swagger

A página exibe todos os endpoints disponíveis, seus parâmetros, formatos de entrada e saída, além de exemplos de requisições. A interface também permite testar as rotas diretamente, usando o botão Authorize, que solicita o token JWT gerado pelo login.

A seguir, são apresentados os principais recursos disponíveis na API, acompanhados de um resumo funcional e exemplos de requisições.

2.10.1. Tabela Geral de Endpoints

A seguir, são apresentados a lista dos principais endpoints que foram criados no projeto Auravet:

Grupo	Método	Endpoint	Descrição
Health	GET	/health	Verifica disponibilidade da API
Auth	POST	/auth/login	Autentica um colaborador interno
Auth	POST	/auth/register	Cadastra um novo colaborador
Auth	GET	/auth/me	Retorna dados do usuário autenticado
Users	GET	/users	Lista colaboradores internos
Users	PATCH	/users/{id}	Atualiza dados de um colaborador
Users	PATCH	/users/{id}/status	Ativa/desativa um colaborador
Roles	GET	/roles	Lista funções cadastradas
Roles	POST	/roles	Cria uma nova função
Roles	DELETE	/roles/{id}	Remove uma função
Owners	GET	/owners	Lista tutores cadastrados
Owners	POST	/owners	Cadastra um tutor
Animals	GET	/animals	Lista animais cadastrados
Animals	POST	/animals	Cadastra um novo animal
Services	POST	/services	Registra um atendimento
Services	GET	/services/catalog	Lista serviços disponíveis

Grupo	Método	Endpoint	Descrição
Services	DELETE	/services/{id}	Remove um atendimento

2.10.2. Fluxo de autorização no Swagger

Para testar rotas protegidas, o Swagger utiliza o modelo Bearer Token. O fluxo funciona assim:

1. Faça login pelo próprio Swagger:

POST /auth/login

The screenshot shows the Swagger UI interface for a POST request to '/auth/login'. The request body is defined as follows:

```
{
  "email": "admin@auravet.com",
  "password": "Admin123!"
}
```

The response section shows a curl command and a request URL:

```
curl -X 'POST' \
  'http://localhost:4000/auth/login' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "email": "admin@auravet.com",
    "password": "Admin123!"
}'
```

Request URL: <http://localhost:4000/auth/login>

Figura 46. Execução do login pelo Swagger

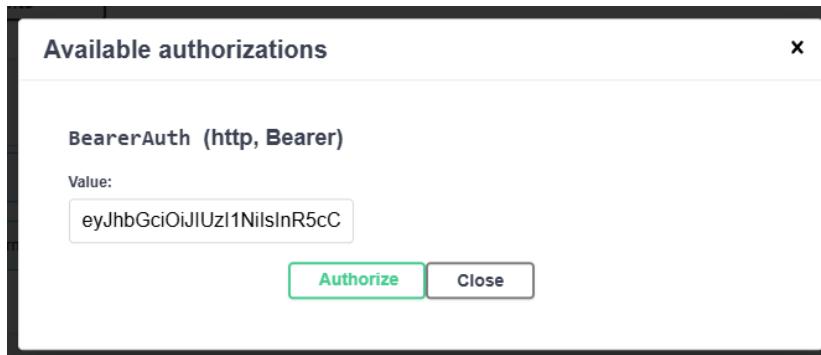
2. Copie o token JWT retornado em "token".

The screenshot shows the Swagger UI interface displaying the response body of the login request. The token is shown in the 'token' field:

```
{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWJ1IjIwM1MjZ1b3a0MDJ1dHRvdjEydt0NzZ21iwiIcn0sZUIkIjoiQURNSUJU1RSQURPUisImIvZHVsZX0h101siYn5pbmFsc3cn0ZStsIaNhC2hpxX6WVnjZKxzIiwb3duZXJ0nJYQ0l1C1vd251cm06d3pdGU1lC1w0c0zpy2hFk1iiv1ch3v2HvjdtM6d3pdGU1lC1c2VycrptWShz2U1XSw1ZKwIjoxNzYz0DY3Mzk0fQ.vNqjHfVtghm03jeXgft5T7G6cGKwt2UXjghQyE",
  "id": "cmlzbGeop002etwv12um176v",
  "name": "Administrador Auravet",
  "email": "admin@auravet.com",
  "roles": [
    {
      "id": "ADMINISTRADOR",
      "slug": "ADMINISTRADOR",
      "name": "Administrador"
    }
  ],
  "isActive": true,
  "lastLoginAt": "2025-11-23T02:09:54.563Z",
  "modules": [
    "animals:read",
    "animals:write",
    "users:read",
    "users:write",
    "owners:read",
    "owners:write",
    "products:read",
    "products:write"
  ]
}
```

Figura 47. Retorno do Token

3. Clique no botão Authorize no topo da página.
4. Cole o token no campo:



5. Agora todas as rotas protegidas ficam liberadas para teste.

Method	Endpoint	Description	Status
POST	/roles	Cria uma nova função	
GET	/roles/modules	Lista módulos disponíveis	
GET	/roles/{id}	Obtém uma função específica	
PATCH	/roles/{id}	Atualiza informações básicas de uma função	
DELETE	/roles/{id}	Remove uma função	
PATCH	/roles/{id}/modules	Atualiza os módulos disponíveis para uma função	
GET	/owners	Lista todos os tutores cadastrados	

Figura 48. Rotas protegidas já liberadas pelo token

2.10.3. Exemplos de Chamadas

Para demonstrar o funcionamento da API, seguem alguns exemplos reais, baseados nos endpoints existentes.

Exemplo: GET /owners (Listar Tutores)

O exemplo abaixo demonstra uma consulta à rota **GET /owners**, responsável por retornar a lista de tutores cadastrados. Esse endpoint exige autenticação prévia – o token deve ser enviado via header `Authorization: Bearer <token>`.

The screenshot shows the Postman interface for the `GET /owners` endpoint. The URL is `/owners` and the description is "Lista todos os tutores cadastrados". The "Parameters" section is empty. The "Responses" section shows a 200 status code response with a media type of `application/json`. The example value is a JSON array representing a tutor object:

```
[
  {
    "id": "string",
    "name": "string",
    "email": "user@example.com",
    "telefone": "string",
    "createdAt": "2025-11-23T02:18:54.118Z"
  }
]
```

Figura 50. Informações da API owners

The screenshot shows the Postman interface displaying the response from the `GET /owners` endpoint. The "Responses" section includes a "Curl" command and a "Request URL" of `http://localhost:4000/owners`. The "Server response" section shows a 200 status code with a "Response body" containing a JSON array of tutor objects. One object is partially visible:

```
[
  {
    "id": "cmib26im0016ttovc19g5x8d",
    "name": "Vicente Elias Miguel da Cunha",
    "email": "vicenteeliasdacunha@plenamenteh.com.br",
    "telefone": "+14927714286",
    "cpf": "00290259827",
    "logradouro": "Rua Sebastião Barreto",
    "numero": "104",
    "complemento": null,
    "bairro": "Jardim Trieste Cavichioli (Padre Nóbrega)",
    "cidade": "Marília",
    "estado": "SP",
    "cep": "17533334",
    "createdAt": "2025-11-23T01:47:58.079Z",
    "animals": [
      {
        "id": "cmib26fc6007ittovekqcwfr8",
        "name": "Zéfiro",
        "especie": "CACHORRO",
        "raça": "Border Collie",
        "nascimento": "2020-12-08T00:00:00.000Z",
        "ownerId": "cmib26im0016ttovc19g5x8d",
        "createdAt": "2025-11-23T01:47:58.103Z",
        "services": [
          {
            "id": "cmib26fc6007ittovekqcwfr8",
            "animalId": "cmib26ejb001ottovag67baq1"
          }
        ]
      }
    ]
  }
]
```

Figura 49. Resposta da API owners

Exemplo: POST /animals (Cadastrar um Pet)

A seguir, mostramos o fluxo de cadastro de um novo animal via POST /animals. Esse exemplo é útil para evidenciar: envio de corpo JSON estruturado; validações aplicadas pelo Zod; retorno com objetos aninhados (animal + tutor); uso do token JWT para autorização.

```
{
  "nome": "Laila",
  "especie": "CACHORRO",
  "raca": "SRD",
  "nascimento": "2021-11-23",
  "ownerId": "cmib26ein#0016ttov619g5x8q"
}
```

Figura 52. Exemplo de Body para post na API animals

```
curl -X 'POST' \
  'http://localhost:4000/animals' \
  -H 'accept: application/json' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJzdWJlOiJjbw1iMjZlb3AwMDJ1dHRvdjEydw10NzZ2Iiwicm9sZUlkIjoiQUtNSU5Ju1RSQURPUiIsIm1vZHVsZXMiOisiYw5pbWFsczpyZhFkIiwiYw5pbWFsczp3cm10ZSIsImNh2h-d'{
  "id": "cmib3gbeq000110ssz9yacy43",
  "nome": "Laila",
  "especie": "CACHORRO",
  "raca": "SRD",
  "nascimento": "2021-11-23",
  "ownerId": "cmib26ein#0016ttov619g5x8q"
}
```

Request URL
http://localhost:4000/animals

Server response

Code	Details
201	Response body

```
{
  "id": "cmib3gbeq000110ssz9yacy43",
  "nome": "Laila",
  "especie": "CACHORRO",
  "raca": "SRD",
  "nascimento": "2021-11-23T00:00:00.000Z",
  "ownerId": "cmib26ein#0016ttov619g5x8q",
  "createdAt": "2025-11-23T02:23:40.218Z",
  "owner": {
    "id": "cmib26ein#0016ttov619g5x8q",
    "name": "Vicente Elias Miguel da Cunha",
    "email": "vicenteeliasdacunha@plenamentebr.com.br",
    "telefone": "+5514992714286",
    "cpf": "00290259827",
    "logradouro": "Rua Sébastião Barreto",
    "numero": "104A",
    "complemento": null,
    "bairro": "Jardim Trieste Cavichioli (Padre Nóbrega)",
    "cidade": "Marília",
    "estado": "SP",
    "cep": "17533334",
    "createdAt": "2025-11-23T01:47:58.079Z"
  },
  "services": []
}
```

Figura 51. Exemplo de resposta ao Post Animals

Exemplo: PATCH /users (Atualizar status de um usuário)

O endpoint **PATCH /users/{id}/status** é utilizado para ativar ou desativar um colaborador. Esse fluxo ilustra: uso de parâmetros na URL ({id}); envio de corpo de atualização parcial; controle de permissões baseado em módulos; resposta detalhada com dados associados ao perfil do colaborador.

PATCH /users/{id}/status Ativa ou desativa um colaborador

Parameters

Name	Description
id * required	string(\$cuid) (path)

Request body required

```
{
  "isActive": false
}
```

application/json

Cancel Reset

Figura 53. Exemplo de Patch de Users

Responses

Curl

```
curl -X 'PATCH' \
  'http://localhost:4000/users/cmib26ezd0044ttovbxeps2w/status' \
  -H 'accept: application/json' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJzdWJiOiJjbWl0MjZlbnM0J1dHlRvdjEydwI0NzZ2Iiwicm9sZUlkiJoiQUFRNSU5JUR5QURPUiIiM1vZHsZXMiOlsiyW5pbmFsczpyZWFKiwiY5pbmFsczp3cm10ZSIsImNhczIi'
  -d '{"isActive": false}'
```

Request URL

http://localhost:4000/users/cmib26ezd0044ttovbxeps2w/status

Server response

Code Details

200 Response body

```
{
  "user": {
    "id": "cmib26ezd0044ttovbxeps2w",
    "name": "Rodrigo Faria Montenegro",
    "email": "rodrigo.montenegro@auravet.com.br",
    "role": {
      "id": "CONTADOR",
      "slug": "CONTADOR",
      "name": "Contador"
    },
    "isActive": false,
    "lastLoginAt": null,
    "modules": [
      "cashier:access",
      "products:read",
      "services:read"
    ],
    "createdAt": "2025-11-23T01:47:58.681Z",
    "updatedAt": "2025-11-23T14:03:48.826Z",
    "collaboratorProfile": {
      "especialidade": "Controladoria e Planejamento Financeiro",
      "crmv": null,
      "turnos": [
        "MANHA",
        "TARDE"
      ]
    }
  }
}
```

Figura 54. Exemplo de resposta do PATCH de usuários

Exemplo: PUT /animals (Atualizar um Pet)

Este endpoint permite editar os dados de um animal já cadastrado. No Swagger, basta informar o ID no path, ajustar o JSON no corpo da requisição e executar a chamada autenticada. A resposta retorna o objeto atualizado, confirmando a alteração imediatamente.

The screenshot shows the Swagger UI interface for a PUT request to the endpoint `/animals/{id}`. The path parameter `{id}` is set to `cmb3gbeg000110ssz9yacy43`. The request body is defined as `application/json` and contains the following JSON:

```
{
  "nome": "Laila",
  "especie": "CACHORRO",
  "raca": "Akita",
  "nascimento": "2023-10-11",
  "ownerId": "cmb26eim0016ttov619g5x8q"
}
```

Figura 56. Exemplo de PUT na API Animals (Body)

The screenshot shows the Swagger UI interface for the same PUT request. It includes a 'Responses' section with a 'curl' example:

```
curl -X 'PUT' \
'http://localhost:4000/animals/cmb3gbeg000110ssz9yacy43' \
-H 'accept: application/json' \
-H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eJzdMIIoijbWlMiMjZl5AwMDJldHRvdjEydwI0NzZ2Iiwicm9sZUlkjoiQURNSU5JU1RSQURPUiisImIvZHVsZXMiOisiYw5pbWFsczpyZWFrIiwiYw5pbWFsczpzcm10ZSIiMNh2H
-d' {
  "nome": "Laila",
  "especie": "CACHORRO",
  "raca": "Akita",
  "nascimento": "2023-10-11",
  "ownerId": "cmb26eim0016ttov619g5x8q"
}'
```

Below it is a 'Request URL' field containing `http://localhost:4000/animals/cmb3gbeg000110ssz9yacy43`. The 'Server response' section shows the status code 200 and the response body:

```
{
  "id": "cmb3gbeg000110ssz9yacy43",
  "nome": "Laila",
  "especie": "CACHORRO",
  "raca": "Akita",
  "nascimento": "2023-10-11T00:00:00.000Z",
  "ownerId": "cmb26eim0016ttov619g5x8q",
  "createdAt": "2025-11-23T02:23:40.218Z",
  "owner": {
    "id": "cmb26eim0016ttov619g5x8q",
    "nome": "Vicente Elias Miguel da Cunha",
    "email": "vicenteeliasdacunha@elenamenteh.com.br",
    "telefone": "(149) 2774286",
    "cep": "00390298272",
    "logradouro": "Rua Sebastião Barreto",
    "numero": "104",
    "complemento": null,
    "bairro": "Jardim Trieste Cavichioli (Padre Nóbrega)",
    "cidade": "Marília",
    "estado": "SP",
    "cep": "17533334",
    "createdAt": "2025-11-23T01:47:58.079Z"
}
```

Figura 57. Resposta do PUT Animals

2.10.4. Modelos de resposta e códigos de erro

Todos os modelos utilizados pela API estão documentados automaticamente no Swagger. Foram realizadas algumas adaptações na linguagem para atender a necessidade da interface do usuário. Porém, os códigos padrão são:

Código	Significado
200	Sucesso
201	Recurso criado
400	Payload inválido
401	Falha de autenticação (token ausente ou inválido)
403	Usuário sem permissão
404	Recurso não encontrado
500	Erro interno inesperado

Abaixo um exemplo da documentação da resposta no Swagger:

The screenshot shows the Swagger UI interface for a 'Responses' section. It lists a single entry for 'Code: 201' with the description 'Animal criado'. Below the description, there's a dropdown menu labeled 'Media type' set to 'application/json', with a note 'Controls Accept header'. Underneath, there are 'Example Value' and 'Schema' buttons. A large black rectangular box covers the JSON example, which is supposed to show a response body for creating an animal.

Responses		Links
Code	Description	
201	Animal criado	No links

Media type
application/json
Controls Accept header

Example Value | Schema

```
{
  "id": "string",
  "name": "string",
  "especie": "CACHORRO",
  "raça": "string",
  "nascimento": "2025-11-23T02:23:40.387Z",
  "ownerId": "string",
  "createdAt": "2025-11-23T02:23:40.387Z",
  "owner": {
    "id": "string",
    "name": "string",
    "email": "user@example.com",
    "telefone": "string",
    "createdAt": "2025-11-23T02:23:40.387Z"
  }
}
```

Figura 58. Exemplo de resposta no Swagger

A documentação disponibilizada no Swagger consolida todas as rotas da plataforma Auravet em um formato claro, navegável e sempre atualizado, permitindo que qualquer desenvolvedor explore os endpoints, visualize exemplos

reais de requisição/resposta e execute chamadas autenticadas diretamente pela interface.

Os exemplos incluídos nesta seção – contemplando operações **GET**, **POST**, **PUT** e **PATCH** – demonstram na prática o funcionamento da API e destacam o cuidado com validação, respostas padronizadas e consistência dos dados.

Assim, o Swagger atua como a **fonte oficial de consulta técnica** do backend, reduzindo ambiguidades, acelerando o onboarding de novos desenvolvedores e garantindo previsibilidade na integração entre serviços.

2.11. Telas

A interface da Auravet foi desenvolvida para ser direta, limpa e alinhada à identidade visual da marca. Todas as telas seguem o mesmo padrão de cores, tipografia e espaçamento, criando uma experiência consistente tanto para recepção, veterinários, enfermeiros e equipe administrativa.

A navegação lateral organiza os módulos de forma clara e previsível, permitindo que o usuário veja rapidamente as principais funcionalidades disponíveis no sistema. As telas também foram pensadas para reduzir esforço cognitivo, apresentando apenas o essencial de cada etapa e guiando naturalmente o fluxo de trabalho da clínica.

A seguir, apresentam-se as principais telas do sistema, acompanhadas de breves descrições e seus respectivos propósitos.

2.11.1. Tela de Login

A primeira experiência do colaborador com o sistema acontece na tela de login, projetada para transmitir acolhimento e simplicidade – dois pilares fundamentais da identidade Auravet. O layout é limpo, centrado e com foco total no formulário principal, reforçando a ideia de que o acesso ao cuidado começa com segurança e fluidez.

O usuário entra utilizando seu e-mail corporativo e senha interna, recebidos a partir do administrador responsável pela gestão de colaboradores. Textos auxiliares discretos orientam quem ainda não possui cadastro, mantendo o tom empático e

humano característico da marca. A paleta em verde-sálvia e o botão de ação em verde profundo reforçam a presença da marca e criam uma sensação de confiança logo no primeiro contato. Essa tela não só autêntica o usuário – ela dá boas-vindas ao ecossistema Auravet.

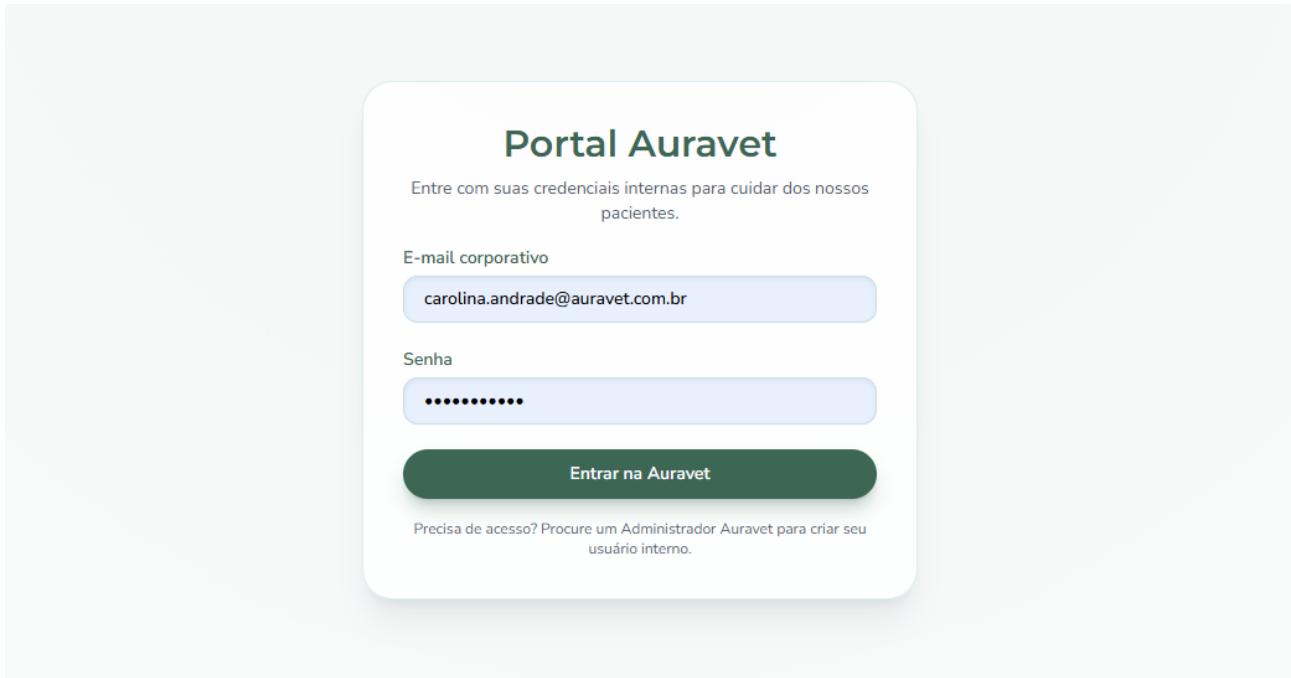


Figura 59. Tela de login do Sistema Auravet

2.11.2. Tela inicial (Dashboard)

A tela inicial da Auravet foi projetada para receber o colaborador de forma acolhedora e objetiva, reforçando o tom humano e sustentável da marca. Logo ao entrar, o usuário encontra uma saudação personalizada, acompanhada de um pequeno texto institucional que reafirma o propósito da clínica: cuidar com ciência, carinho e organização. Essa mensagem funciona como um lembrete constante da missão da Auravet e da importância do trabalho de cada membro da equipe.

A navegação lateral apresenta todos os módulos essenciais – Tutores, Animais, Atendimentos, Agenda, Serviços, Produtos, Financeiro e Caixa – organizados em uma coluna fixa que facilita o deslocamento rápido entre diferentes áreas do sistema. O design clean, as cores suaves e a iconografia minimalista reforçam a experiência fluida e a identidade visual da marca.

No centro da tela, quatro painéis principais apresentam um resumo em tempo real das informações mais relevantes: total de tutores cadastrados, número de pets registrados, indicadores de atendimentos e status do estoque e do catálogo de produtos. Cada card foi pensado para dar ao colaborador uma visão executiva imediata – permitindo identificar pendências, fluxos de trabalho e demandas urgentes com poucos segundos de leitura.

O botão “Atendimento”, em destaque no canto superior, permite registrar um novo atendimento a qualquer momento, reforçando o foco da plataforma no cuidado contínuo e na rapidez operacional.

Essa tela funciona como o “coração operacional” da Auravet: simples, clara, acolhedora e eficiente – tudo para que a equipe possa concentrar energia onde realmente importa: no bem-estar dos pacientes.

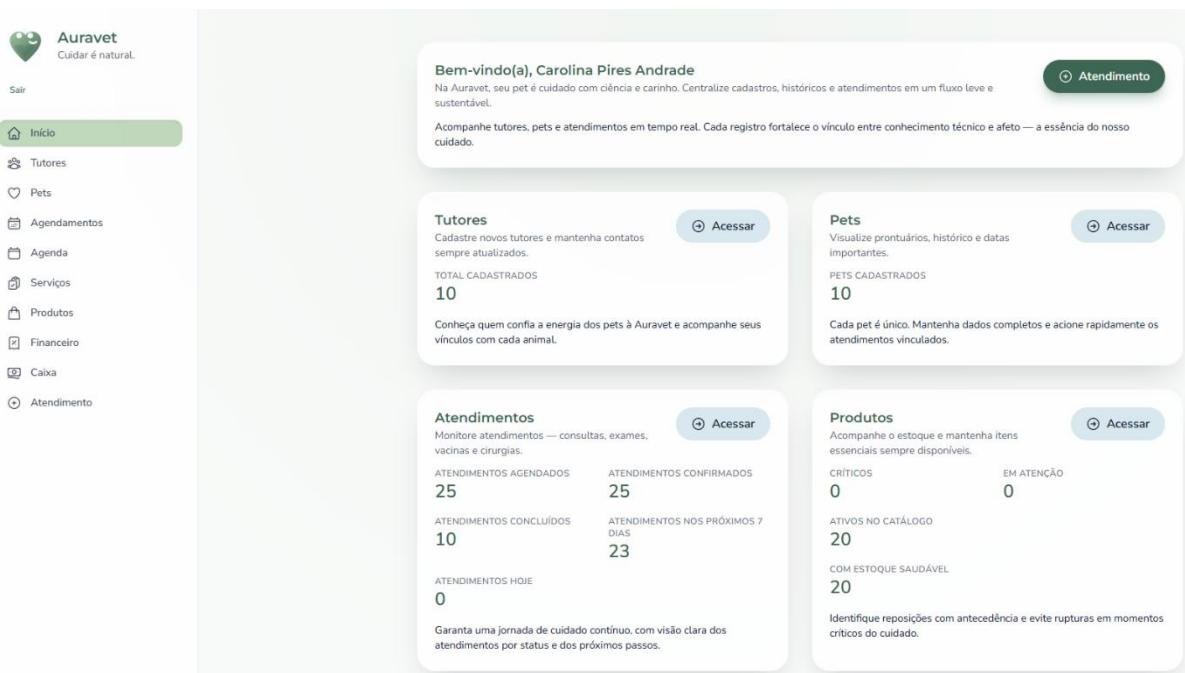


Figura 60. Tela inicial do Sistema Auravet

2.11.3. Tutores

O módulo de Tutores reúne todas as informações relacionadas aos responsáveis pelos animais cadastrados na Auravet. A proposta visual segue o design system da

marca, com foco em leveza, acessibilidade e acolhimento – transmitindo o mesmo cuidado que a clínica aplica na relação com os pets de seus pacientes.

Tela de listagem de tutores

Nome do tutor	E-mail	CPF
Vicente Elias Miguel da Cunha vicenteeliasdacunha@plenamentehr.com.br 14992714286 CPF: 002.902.598-27 Rua Sebastião Barreto, 104 • Jardim Trieste Cavichioli (Padre Nóbrega) • Marília - SP • CEP 17533-334 2 ANIMAIS SOB CUIDADO	email@auravet.com	000.000.000-00
Nair Adriana Caldeira nair_caldeira@focusdm.com.br 14997774000 CPF: 199.513.588-70 Rua Carolina Moraes Almeida, 545 • Senador Salgado Filho • Marília - SP • CEP 17502-290 1 ANIMAIS SOB CUIDADO		
Stella Larissa Vieira stella.larissa.vieira@br.ibn.com 14969882589 CPF: 350.880.078-41 Rua Tadaiti Ishida, 877 • Parque das Indústrias • Marília - SP • CEP 17519-720 1 ANIMAIS SOB CUIDADO		

Figura 61. Tela de listagem de tutores

A tela inicial do módulo apresenta uma visão organizada de todos os tutores cadastrados no sistema. No topo, filtros rápidos permitem buscar responsáveis por nome, e-mail ou CPF, facilitando a navegação mesmo em bases extensas. O layout é limpo e prioriza a leitura: cada cartão exibe dados essenciais de contato, endereço, documentos e a quantidade de animais sob cuidado daquele tutor.

Os botões **Editar** e **Remover** ficam posicionados de maneira discreta, seguindo o estilo minimalista da Auravet, mas permanecem acessíveis para ações administrativas. A opção **Exportar para Excel** amplia a praticidade, permitindo que a equipe gere relatórios ou listas personalizadas quando necessário. O botão **Novo tutor**, localizado no canto superior direito, abre o fluxo de cadastro de forma imediata, reforçando o conceito de navegação intuitiva.

Tela de cadastro de novo tutor

Novo tutor Auravet

Preencha os dados de contato para fortalecer a comunicação empática com cada família.

Nome
Nome completo

E-mail
email@auravet.com

Telefone
(00) 00000-0000

CPF
000.000.000-00

Logradouro
Rua e número

Número Número	Complemento Apto, bloco...
------------------	-------------------------------

Bairro
Bairro

Cidade Cidade	Estado UF
------------------	--------------

CEP
00000-000

Cadastrar

Figura 62. Tela de novo Tutor

Ao cadastrar um novo responsável, o sistema apresenta um formulário completo, distribuído com espaçamento confortável e uso consistente da paleta verde-sálvia da Auravet. Cada input orienta o usuário com placeholders claros, garantindo um preenchimento rápido e sem ambiguidades.

O objetivo dessa tela é oferecer uma experiência fluida, onde a equipe consiga registrar dados de forma segura: nome, telefone, e-mail, CPF, endereço completo e demais elementos que fortalecem a relação entre clínica e família. O botão Cadastrar

finaliza o processo, enquanto a opção Cancelar permite interromper o fluxo sem perder informações.

Tela de edição de tutor

Editar tutor

Preencha os dados de contato para fortalecer a comunicação empática com cada família.

Nome	Vicente Elias Miguel da Cunha	
E-mail	vicenteeliasdacunha@plenalementerh.com.br	
Telefone	14992714286	
CPF	00290259827	
Logradouro	Rua Sebastião Barreto	
Número	104	Complemento Apto, bloco...
Bairro	Jardim Trieste Cavichioli (Padre Nóbrega)	
Cidade	Marília	Estado SP
CEP	17533334	

Cancelar **Salvar alterações**

Figura 63. Tela para editar tutor

A tela de edição segue o mesmo padrão visual do cadastro, porém já pré-preenchida com todos os dados do tutor selecionado. Esta escolha reforça o princípio de eficiência: o colaborador pode ajustar rapidamente qualquer campo, desde telefone até endereço ou informações complementares.

Mensagens claras e tipografia bem definida ajudam o usuário a revisar e atualizar os dados com confiança. O botão **Salvar alterações** assume a cor verde escura institucional, reforçando a ação de confirmação, enquanto o **Cancelar** mantém o fluxo seguro para usuários que desejam apenas consultar uma informação.

Cartão individual do Tutor



Figura 64. Cartão com as principais informações do tutor

Além das telas de listagem e formulários, cada tutor é apresentado em um cartão individual que organiza suas informações essenciais de forma visualmente agradável. Esse cartão destaca nome completo, e-mail, telefone, CPF e endereço completo, além do contador de animais sob sua responsabilidade.

Esses cartões reforçam a lógica de navegação humanizada da Auravet: cada responsável é tratado como parte importante da rede de cuidado dos pets, e a interface traduz esse vínculo com clareza, delicadeza e informação acessível.

2.11.4. Pets

No módulo de Pets, concentram-se todas as informações essenciais sobre os animais acompanhados pela Auravet. Cada pet carrega sua própria história clínica, seus vínculos familiares e um conjunto de dados que orientam o cuidado diário da equipe.

O objetivo dessa área do sistema é oferecer uma visão clara, organizada e rápida, permitindo que profissionais acessem tudo o que precisam sem interrupções ou navegações excessivas.

Tela Pets Acompanhados

Pets
Cada pet carrega uma história. Visualize tutores, espécie e histórico de atendimentos em um só lugar.

Pets acompanhados

Toque em um pet para abrir o histórico completo.

Buscar por pet Tutor Espécie Exportar lista

Nome do pet Nome do tutor Todas

Corvo
Gato • Tutor(a): Calebe Julio Rodrigues
BOMBAY

Ônix
Gato • Tutor(a): Nair Adriana Caldeira
BENGAL

Nebulosa
Gato • Tutor(a): Luna Betina Emily Martins
PERSA

Histórico do pet
Atendimentos realizados com datas e valores.
Selecione um pet na lista ao lado para visualizar o histórico de carinho clínico.

Registrar Atendimento Novo Pet

Figura 65. Tela de Pets Acompanhados

A tela inicial do módulo apresenta todos os animais cadastrados e acompanhados pela Auravet. Ela permite localizar rapidamente um pet por nome, tutor ou espécie, oferecendo uma visão organizada dos vínculos e das características de cada animal. É o ponto de partida para acessar o histórico completo, editar registros e registrar novos atendimentos. Ao selecionar um pet, sua ficha detalhada é exibida à direita, garantindo que a navegação seja direta, fluida e totalmente centrada no cuidado do animal.

Tela Novo Pet

Esta tela foi criada para tornar o cadastro de um novo animal simples, acolhedor e rápido. Nela, o profissional preenche os dados principais do pet, como nome, espécie, raça e data de nascimento, além de escolher o tutor responsável. O texto introdutório reforça que, quanto mais completos forem esses dados, mais personalizada será a jornada de bem-estar do animal dentro da clínica. Após o preenchimento, basta confirmar o cadastro para que o pet passe a integrar a lista geral.

Novo pet Auravet

Quanto mais detalhes, mais personalizada fica a jornada de bem-estar do animal.

Nome	<input type="text" value="Nome do pet"/>
Espécie	<input type="text" value="Cachorro"/>
Raça	<input type="text" value="Opcional"/>
Data de nascimento	<input type="text" value="dd/mm/aaaa"/> (calendário)
Tutor	<input type="text" value="Selecione um tutor"/>

Cancelar
Cadastrar pet

Figura 66. Cadastro de novo Pet

Tela Editar Pet

Editar pet

Quanto mais detalhes, mais personalizada fica a jornada de bem-estar do animal.

Nome	<input type="text" value="Corvo"/>
Espécie	<input type="text" value="Gato"/>
Raça	<input type="text" value="Bombay"/>
Data de nascimento	<input type="text" value="04/04/2015"/> (calendário)
Tutor	<input type="text" value="Calebe Julio Rodrigues"/>

Cancelar
Salvar

Figura 67. Tela de edição de Pet

Quando é necessário ajustar informações já cadastradas, a tela de edição apresenta todos os dados do pet de forma clara e organizada. O profissional pode alterar nome, espécie, raça, tutor ou data de nascimento, mantendo o histórico sempre atualizado. Essa tela garante que mudanças familiares, revisões cadastrais e correções rápidas possam ser feitas de forma segura, preservando o vínculo entre animal e tutor.

Tela de Histórico do Pet (Painel Lateral)

Histórico do pet
Atendimentos realizados com datas e valores.

Ônix
Gato • Tutor(a): Nair Adriana Caldeira
NASCIMENTO: 08/03/2020

Contato do tutor
nair_caldeira@focusdm.com.br
14997774000
CPF: 199.513.588-70
Rua Carolina Moraes Almeida, 545 • Senador Salgado Filho • Marília - SP • CEP 17502-290

Abrir histórico completo

Início	Fim	Status
dd/mm/	dd/mm/	Todos

CONSULTA
22/11/2025 • R\$ 180.00
Serviço Consulta clínica geral realizado durante o agendamento seed.
[Editar](#)

Ao selecionar um pet na lista principal, o sistema abre à direita um painel detalhado com informações essenciais do animal. Essa visualização reúne dados como espécie, raça, tutor responsável, contatos e endereço. Logo abaixo, o histórico de atendimentos aparece em ordem cronológica, com valores, datas e descrições. O profissional pode aplicar filtros por período e status, encontrando qualquer atendimento com facilidade. Esse painel permite uma navegação dinâmica, sem troca de páginas, mantendo o fluxo de trabalho ágil.

Figura 68. Painel lateral de Histórico do Pet

Tela Atendimentos do Pet

[Voltar](#)

Atendimentos do pet

Histórico consolidado do pet com filtros rápidos para encontrar atendimentos específicos.

Pet
Dados principais para confirmar que você está no prontuário correto.

Ônix
GATO • Bengal
NASCIMENTO: 08/03/2020

Tutor
Contatos rápidos do responsável pelo pet.
Nair Adriana Caldeira
nair_caldeira@focusdm.com.br
14997774000
CPF: 199.513.588-70
Rua Carolina Moraes Almeida, 545 • Senador Salgado Filho • Marília - SP • CEP 17502-290

Histórico de atendimentos
Veja detalhes rápidos e acesse o atendimento completo.

Consulta CONCLUÍDO
22/11/2025, 12:00:00
R\$ 180,00
Responsável: Dr. Henrique Sales Pacheco
Agendamento: 22/11/2025, 11:00:00 - 22/11/2025, 12:00:00
Serviço Consulta clínica geral realizado durante o agendamento seed.

Filtros
Refine o histórico por período e status.

Status: Todos

De: dd/mm/aaaa Até: dd/mm/aaaa

Limpar filtros

Figura 69. Tela para Atendimentos do Pet

Esta tela consolida a jornada clínica do animal em uma estrutura clara e objetiva. À esquerda, são exibidos os detalhes principais do pet e seu tutor, permitindo que o profissional valide rapidamente se está no prontuário correto. Ao lado, o histórico de atendimentos aparece com destaque, trazendo valores, horários e o responsável pelo atendimento. Um botão permite visualizar ou editar cada atendimento, garantindo que ajustes clínicos ou administrativos sejam feitos com segurança e rapidez. Filtros por status e período completam a tela, facilitando a busca por consultas específicas.

2.11.5. Agendamentos

O módulo de Agendamentos reúne toda a logística clínica da Auravet em um único ambiente, permitindo que a equipe visualize, filtre, organize e conduza consultas e procedimentos com precisão. Ele funciona como o centro operacional da rotina diária, exibindo desde atendimentos futuros até consultas concluídas, com indicadores claros e ações rápidas em cada cartão. Sua estrutura é ampla, mas intuitiva, tornando fácil navegar entre filtros, reagendamentos, confirmações e finalização dos atendimentos.

Tela principal dos Agendamentos

Ao acessar o módulo, o usuário encontra um painel completo dedicado à agenda da clínica. No topo da página, há um conjunto de filtros que permite personalizar a exibição. É possível filtrar por status (Agendado, Confirmado, Concluído), colaborador responsável, tutor, pet, além de selecionar intervalos de data por período inicial e final. Esses filtros garantem uma visualização mais dirigida e ajudam tanto na operação diária quanto na análise retroativa dos atendimentos.

Figura 70. Tela de agendamentos

Após aplicar os filtros, a agenda clínica é apresentada em formato de cartões, cada um representando um atendimento individual. Esses cartões reúnem informações essenciais, como horário, duração, tipo de atendimento, tutor, responsável clínico, assistente (quando houver), endereço do tutor e observações registradas no momento da criação do agendamento. Tudo é organizado de maneira clara e fácil de interpretar, de forma que o usuário possa tomar decisões rápidas sem abrir telas adicionais.

Tela Novo Agendamento

Ao clicar em Novo agendamento, o usuário é direcionado a um formulário completo para registrar uma nova consulta. Nesse formulário, é possível selecionar o tutor, escolher o pet, definir o profissional responsável, o assistente de apoio, o tipo de atendimento e preencher horário de início e fim.

Novo agendamento

Selecione tutor, pet e equipe para registrar um novo atendimento.

Tutor

Selezione um tutor

Pet

Selezione um pet

Profissional responsável

Selezione um colaborador

Profissional assistente

Sem assistente

Tipo de atendimento

Selezione o tipo de atendimento

Início

dd/mm/aaaa --:--

Fim

dd/mm/aaaa --:--

Status

Agendado

Observações (opcional)

Anotações internas, orientações ao tutor ou observações clínicas

Criar agendamento

Figura 71. Tela para criar agendamento

O formulário também exibe um campo opcional para observações internas, que podem incluir instruções prévias, lembretes ou qualquer anotação clínica relevante. Todos os campos são apresentados com validação amigável, garantindo que só informações consistentes sejam enviadas. Após a confirmação, a consulta aparece imediatamente na agenda com todas as informações preenchidas.

Ações da Agenda Clínica

Dentro da tela de Agendamentos, cada cartão de consulta traz, além dos dados clínicos e administrativos do atendimento, um conjunto de ações rápidas que facilitam o fluxo diário da equipe. Esses botões funcionam como atalhos para tomada de decisão, reduzindo navegação e agilizando atividades operacionais.



Figura 72. Ações da agenda clínica

Reagendar atendimento

Ao clicar em Reagendar, o usuário abre imediatamente um pequeno painel dedicado à alteração de data e horário.

Essa interface aparece diretamente dentro do próprio cartão da consulta, sem redirecionar para outra página, o que torna o processo leve e objetivo.

Eco • Agendado
24/11/2025, 06:00 — 24/11/2025, 07:00 • Duração: 60 min
Tipo de atendimento: Tipo não informado
Tutor(a): Stella Larissa Vieira • Veterinário(a): Dra. Camila Teixeira Lins
14989882589
CPF: 350.880.078-41
Rua Tadaiti Ishida, 877 • Parque das Indústrias • Marília - SP • CEP 17519-720
Assistente: Enf. Larissa Melo Coutinho

Veterinário disponível Assistência liberada

Observações: Agendamento criado via seed (check-up / consulta rotineira).

Início	Fim	Notas
24/11/2025 06:00	24/11/2025 07:00	Agendamento criado via seed

Salvar novo horário Cancelar

Figura 73. Tela para reagendar o atendimento

O formulário já vem preenchido com o horário original, permitindo ao colaborador apenas ajustar os campos “Início” e “Fim”. Após definir o novo período, basta confirmar para atualizar a agenda. Essa ação é útil principalmente em situações de conflito de horário, solicitações do tutor ou indisponibilidade repentina do profissional.

A mudança é aplicada de forma segura e o agendamento permanece com o mesmo ID, histórico e vínculo com o tutor e o pet, preservando toda a rastreabilidade do atendimento.

Confirmar Presença

O botão Confirmar presença aparece nos agendamentos ainda marcados como agendado.

Zéfiro • Confirmado
24/11/2025, 06:00 — 24/11/2025, 07:00 • Duração: 60 min
Tipo de atendimento: Tipo não informado
Tutor(a): Vicente Elias Miguel da Cunha • Veterinário(a): Dr. Lucas Almeida Furtado
14992714286
CPF: 002.902.598-27
Rua Sebastião Barreto, 104 • Jardim Trieste Cavichioli (Padre Nóbrega) • Marília - SP • CEP 17533-334
Assistente: Enf. Thiago Ramos Silveira

Veterinário disponível Assistência liberada

Observações: Agendamento confirmado via seed (consulta já confirmada com tutor).

Figura 74. Atendimento confirmado

Ao acioná-lo, o sistema altera o status para Confirmado, permitindo que a clínica diferencie consultas apenas marcadas daquelas realmente confirmadas pelos tutores.

Essa simples mudança de status garante melhor organização da agenda diária e ajuda a equipe a prever demanda real, reduzindo ausências e otimizando o tempo dos profissionais.

O cartão do agendamento é atualizado instantaneamente, incluindo a etiqueta de status visual que destaca a consulta como confirmada.

Registrar atendimento

Quando o colaborador seleciona Registrar atendimento, ele é levado à tela de atendimento clínico – já com:

- o tutor selecionado,
- o pet relacionado,
- o agendamento vinculado,
- e com a data já preenchida.

Essa pré-carga de dados reduz retrabalho e minimiza erros de digitação, permitindo que o profissional foque apenas no conteúdo clínico do atendimento.

Registrar Atendimento

Combine múltiplos serviços do catálogo e produtos utilizados para gerar um atendimento completo.

[Abrir catálogo de serviços](#)

Dados do atendimento

Preencha com atenção para manter o histórico impecável e inclua quantos serviços forem necessários no mesmo atendimento.

Agendamento (opcional)

24/11/2025 06:00 - 07:00 • Zéfiro (Vicente Elias Miguel da Cunha)

Pet	Tutor
Zéfiro — Tutor(a): Vicente Elias Miguel da Cunha	Vicente Elias Miguel da Cunha

Tutor definido automaticamente a partir do pet selecionado.

Início do atendimento	Término do atendimento
24/11/2025 06:00	24/11/2025 07:00

Opcional: preencha ao concluir para registrar a duração.

Profissional responsável	Profissional assistente
Dr. Henrique Sales Pacheco — henrique.pacheco@auravet.com.br	Enf. Thiago Ramos Silveira — thiago.silveira@auravet.com.br

Serviços do catálogo

Selezione um ou mais serviços do catálogo para registrar o que foi realizado no atendimento.

Adicionar serviço

Produtos utilizados

Registre os insumos aplicados durante o atendimento para manter o estoque sempre atualizado.

Adicionar produto

Figura 75. Redirecionamento ao clicar em Registrar Atendimento

Depois que o atendimento é finalizado nessa tela, ele passa a constar no histórico do pet com status Concluído, incluindo informações de valores, horários e anotações clínicas.

Gerar PDF do atendimento

Para agendamentos que já foram concluídos, a agenda exibe o botão Gerar PDF do atendimento.

Agenda clínica

Confirme, reagende ou finalize consultas com indicadores de disponibilidade.

Luna • Concluído	Gerar PDF do atendimento
13/11/2025, 11:00 — 13/11/2025, 12:00 • Duração: 60 min	
Tipo de atendimento: Tipo não informado	
Tutor(a): Alice Sophie Simone Nogueira • Veterinário(a): Dra. Bianca Correia Menezes	
14999580639	
CPF: 363.918.978-75	
Rua Leonel de Souza Barros, 190 • Jardim Tangará • Marília - SP • CEP 17516-022	
Assistente: Enf. Paula Regina Saldanha	
Veterinário disponível	Assistência liberada
Observações: Atendimento de Consulta de retorno concluído via seed.	

Figura 76. Atendimento concluído é possível gerar PDF

Esse recurso exporta um documento completo com:

- identificação do pet e tutor,
- dados do profissional responsável,
- horário e duração,
- descrição do atendimento,
- observações,
- e informações administrativas.

O PDF é ideal para impressão, envio ao tutor, arquivamento e compliance documental.



Auravet

Ficha de atendimento

Atendimento de Golias em 13 de novembro de 2025

Tipo de atendimento: Outros cuidados

Data do atendimento: 13/11/2025, 12:00:00

Tutor

Nome: Victor Henry da Rosa

E-mail: victor.henry.darosa@live.ie

Telefone: 14994781561

CPF: 959.751.988-79

Endereço: Rua Anelda Volta Brazini, 343 • Antonio Carlos Nascimento da Silva • Marília - SP • CEP 17523-893

Pet

Nome: Golias

Espécie: CACHORRO

Raça: Rottweiler

Horários do atendimento

Início: 13/11/2025, 11:00

Término: 13/11/2025, 12:00

Serviços realizados

Itens: Nenhum serviço cadastrado

Produtos utilizados

1. Kit Curativo Pequeno:

Quantidade: 1

Subtotal: R\$ 54,90

Totais do atendimento

Serviços: R\$ 0,00

Produtos: R\$ 54,90

Total geral: R\$ 130,00

Figura 77. Print do PDF Ficha de Atendimento

A geração é imediata e ocorre diretamente na tela da agenda clínica, sem a necessidade de abrir o atendimento completo.

Cancelar Agendamento

O cancelamento de um agendamento faz parte da rotina real da clínica e foi pensado para acontecer de forma clara, segura e rastreável. Na própria tela de agendamentos, cada cartão de consulta exibe, à direita, um conjunto de ações que inclui o botão “Cancelar agendamento”. Quando essa opção é acionada, o sistema solicita uma confirmação rápida e, em seguida, atualiza o status do compromisso, liberando automaticamente o horário na agenda do médico e da equipe de apoio. O registro do agendamento não desaparece: ele permanece associado ao tutor e ao pet, servindo como histórico de planejamento, mas claramente marcado como cancelado.



Figura 78. Card de agendamento com botão de cancelamento

Essa decisão de manter o registro, e não o apagar, reforça o compromisso da Auravet com transparência e rastreabilidade. A equipe consegue compreender facilmente por que um horário ficou vago, quem realizou o cancelamento e quais foram as informações envolvidas. O botão “Registrar atendimento” permanece disponível apenas para agendamentos ativos, reforçando a lógica de que o cancelamento é uma escolha consciente de gestão do fluxo clínico.

Status e conflito de agenda

Dentro do módulo de Agendamentos, além dos status tradicionais como Agendado, Confirmado e concluído, existe também uma sinalização automática de conflito de agenda. Esse alerta aparece quando dois ou mais atendimentos coincidem no mesmo horário envolvendo profissionais semelhantes – seja o veterinário principal, seja o assistente clínico.

Zéfiro • Agendado

29/11/2025, 11:00 — 29/11/2025, 12:00 • Duração: 60 min
Tipo de atendimento: Tipo não informado
Tutor(a): Vicente Elias Miguel da Cunha • Veterinário(a): Dra. Bianca Correia Menezes 14992714286
CPF: 002.902.598-27
Rua Sebastião Barreto, 104 • Jardim Trieste Cavichioli (Padre Nóbrega) • Marília - SP • CEP 17533-334
Assistente: Enf. Thiago Ramos Silveira

Veterinário com conflito **Assistência liberada**

Observações: Agendamento confirmado via seed (consulta já confirmada com tutor).

Confirmar presença **Reagendar** **Registrar atendimento**

Luna • Confirmado

29/11/2025, 11:00 — 29/11/2025, 12:00 • Duração: 60 min
Tipo de atendimento: Tipo não informado
Tutor(a): Alice Sophie Simone Nogueira • Veterinário(a): Dra. Bianca Correia Menezes 14999580639
CPF: 363.918.978-75
Rua Leonel de Souza Barros, 190 • Jardim Tangará • Marília - SP • CEP 17516-022
Assistente: Enf. Gustavo Vieira Campos

Veterinário com conflito **Assistência liberada**

Observações: Agendamento confirmado via seed (consulta já confirmada com tutor).

Reagendar **Registrar atendimento**

Figura 79. Exemplo de veterinário em conflito

Quando isso acontece, a interface exibe um selo visual destacado, como “Veterinário com conflito” ou “Assistência com conflito”, indicando exatamente qual membro da equipe está associado ao choque de horários. Essa sinalização permite que o usuário identifique rapidamente onde está o problema e possa agir sem precisar revisar toda a agenda manualmente.

Trovão • Agendado

25/11/2025, 07:00 — 25/11/2025, 08:00 • Duração: 60 min
Tipo de atendimento: Tipo não informado
Tutor(a): Kauê Carlos Eduardo Silva • Veterinário(a): Dra. Bianca Correia Menezes
14998555824
CPF: 162.036.568-52
Rua Antonio José da Silva, 772 • Vereador Eduardo Andrade Reis • Marília - SP • CEP 17526-710
Assistente: Enf. Vanessa Borges Freire

Veterinário disponível **Assistência com conflito**

Observações: Agendamento criado via seed (check-up / consulta rotineira).

Zéfiro • Confirmado

25/11/2025, 07:00 — 25/11/2025, 08:00 • Duração: 60 min
Tipo de atendimento: Tipo não informado
Tutor(a): Vicente Elias Miguel da Cunha • Veterinário(a): Dr. Lucas Almeida Furtado
14992714286
CPF: 002.902.598-27
Rua Sebastião Barreto, 104 • Jardim Trieste Cavichioli (Padre Nóbrega) • Marília - SP • CEP 17533-334
Assistente: Enf. Vanessa Borges Freire

Veterinário disponível **Assistência com conflito**

Observações: Agendamento confirmado via seed (consulta já confirmada com tutor).

Figura 80. Exemplo de assistente em conflito

Esse comportamento é especialmente importante para manter a organização da clínica, evitando sobrecargas de profissionais, atrasos e perda de qualidade no atendimento. Sempre que um conflito surge, o cartão do agendamento traz o aviso logo abaixo das informações principais, facilitando o reconhecimento imediato da situação.

Agendamentos
Organize consultas, confirme presenças e visualize conflitos de agenda dos colaboradores.

Filtrar agenda
Personalize a visão por status, tutor, pet, responsável clínico ou conflitos de agenda.

Status	Colaborador	Tutor	Pet	De	Até
Todos	Todos os profis	Todos os tutores	Todos os pets	dd/mm/aaaa	dd/mm/aaaa

Conflitos de agenda
Marque as opções para visualizar apenas horários com sobreposição de compromissos e confira os chips de disponibilidade destacados na lista.

Conflito do tutor(a) Conflito da assistente

Aplicar filtros **Limpar**

Agenda clínica
Exibindo apenas horários com conflito, destacados pelos chips de disponibilidade.
Filtro ativo: exibindo agendamentos com conflito do tutor(a).

Ônix • Agendado
26/11/2025, 08:00 — 26/11/2025, 09:00 • Duração: 60 min
Tipo de atendimento: Tipo não informado
Tutor(a): Nair Adriana Caldeira • Veterinário(a): Dra. Camila Teixeira Lins
14997774000
CPF: 199.513.588-70
Rua Carolina Moraes Almeida, 545 • Senador Salgado Filho • Marília - SP • CEP 17502-290
Assistente: Enf. Vanessa Borges Freire

Veterinário com conflito **Assistência liberada**

Observações: Agendamento criado via seed (check-up / consulta rotineira).

Figura 81. Filtros de conflito de agenda

Nos filtros da agenda, o usuário pode marcar **Conflito do tutor(a)** ou **Conflito da assistente** para visualizar apenas os agendamentos onde a equipe está alocada além da disponibilidade.

Ao ativar esses filtros, a lista exibe exclusivamente os compromissos problemáticos, destacados pelos chips de conflito. Isso facilita detectar horários críticos, ajustar a

escala e reagendar atendimentos antes que o fluxo da clínica seja impactado. A interface ainda exibe um aviso indicando que apenas agendamentos em conflito estão sendo mostrados, mantendo a navegação clara e objetiva.

2.11.6. Agenda Inteligente

A Agenda Inteligente oferece uma visão estratégica da ocupação da equipe clínica, combinando informações de disponibilidade, compromissos e capacidade operacional. Ela foi pensada para ajudar a clínica a antecipar demandas, identificar horários críticos e equilibrar a carga entre os profissionais. Conforme o usuário ajusta filtros e períodos, a tela se reorganiza automaticamente e apresenta resumos completos da capacidade e dos agendamentos, sempre de forma clara e visual.

The screenshot displays the 'Agenda inteligente' (Intelligent Agenda) interface. At the top, there's a header with the title 'Agenda inteligente', a link 'Ir para hoje' (Go to today), and a sub-instruction 'Visualize a ocupação diária, semanal ou mensal da equipe clínica com indicadores de capacidade.' (View daily, weekly or monthly occupancy of the clinical team with capacity indicators). Below the header, there's a section titled 'Ajuste o panorama' (Adjust the panorama) with a sub-instruction 'Combine visão temporal, profissional e status em segundos.' (Combine temporal, professional and status views in seconds). It includes filters for 'Visualização' (Weekend), 'Data de referência' (24/11/2025), 'Colaborador' (Dr. Lucas Almeida Furtado), and 'Status' (Todos). Buttons for 'Atualizar' (Update) and 'Limpar filtros' (Clear filters) are also present. The main area is divided into three sections: 'Resumo de capacidade' (Capacity summary), 'Mapa da semana' (Weekly map), and a detailed appointment view for 'Quinta-Feira, 27/11'. The 'Resumo de capacidade' section shows a period from 23/11/2025 to 30/11/2025, a total of 3 appointments (3 confirmed), 0 completed, 2 pending, and 53 available slots (3 occupied of 56). The 'Mapa da semana' section allows users to identify conflicts by day. The detailed view for 'Quinta-Feira, 27/11' shows an appointment from 12:00 to 13:00 for Dr. Calebe Julio Rodrigues, Corvo, at address Avenida Antônio Borella, 116, Jardim São Domingos, Marília - SP, CEP 17514-747. The status is 'Confirmado' (Confirmed). A button 'Registrar atendimento' (Register appointment) is visible at the bottom of this section.

Figura 82. Tela da agenda inteligente

Ajuste de panorama

Nesta área o usuário define como deseja enxergar a agenda: dia, semana ou mês. A escolha da data de referência determina o intervalo exibido e, ao selecionar um colaborador específico, a visualização é ajustada apenas aos seus atendimentos. Filtros de status permitem analisar somente consultas confirmadas, concluídas, pendentes ou todas ao mesmo tempo. A cada atualização, toda a agenda é recalculada e reorganizada, permitindo uma leitura precisa do cenário atual da clínica.

Resumo da capacidade

Esta seção reúne os principais indicadores de uso da agenda dentro do período filtrado. Quando um colaborador é selecionado, o sistema calcula automaticamente sua capacidade total com base nos turnos configurados no perfil (manhã, tarde e noite). Cada turno corresponde a quatro vagas de atendimento por padrão e, caso algum turno não esteja definido, o sistema aplica uma capacidade padrão de fallback. Depois disso, a Agenda Inteligente verifica quantos agendamentos ainda não concluídos o profissional possui – esses são os compromissos ocupados. Com isso, a capacidade livre é determinada pela diferença entre total de vagas e agendamentos pendentes, garantindo uma visão realista da disponibilidade do profissional no período escolhido.



Figura 83. Resumo da capacidade do funcionário

Se nenhum colaborador for filtrado, esses indicadores ficam desativados, permitindo apenas visualizar a quantidade bruta de agendamentos sem projeção de capacidade.

Mapa da Semana (Mês ou Dia)

Logo abaixo, o usuário encontra a visão de distribuição dos agendamentos ao longo dos dias. Neste mapa, cada compromisso aparece com seus dados completos: tutor, pet, responsáveis clínicos, endereço e notas internas. A visualização destaca horários e durações, facilitando a identificação de padrões, picos de demanda ou possíveis conflitos. Esta é a parte da agenda voltada à operação diária, permitindo navegar pelos dias e agir rapidamente sobre cada consulta.

Ações rápidas

Cada agendamento exibido na Agenda Inteligente vem acompanhado de botões de ação que agilizam o dia a dia da equipe. A confirmação de presença permite atualizar o status da consulta com um único clique, enquanto a opção de reagendar abre imediatamente a edição de data e horário, facilitando ajustes de última hora sem sair da tela. O botão “Registrar atendimento” leva o usuário diretamente para a ficha de atendimento, já pré-preenchida com todas as informações do agendamento – como tutor, pet, profissionais responsáveis e horário – reduzindo retrabalho e acelerando o fluxo clínico.



Figura 84. Card da agenda Inteligente

Quando o agendamento já está concluído, surge também a opção de Gerar PDF do atendimento, permitindo exportar o registro completo da consulta de forma rápida e organizada. Essa funcionalidade é útil para arquivos internos, compartilhamentos controlados ou documentação complementar.

2.11.7. Serviços

O módulo de Serviços reúne todos os procedimentos clínicos e operacionais oferecidos pela Auravet. Ele funciona como o catálogo central da clínica, permitindo registrar consultas, exames, cuidados específicos e qualquer tipo de atendimento prestado aos pets. Cada serviço conta com informações essenciais – como tipo, profissional indicado, valor sugerido e descrição detalhada – garantindo padronização, transparência e agilidade tanto para o time administrativo quanto para a equipe clínica.

Nesta seção, é possível criar serviços, atualizar itens existentes e filtrar o catálogo de forma rápida, tornando o fluxo de atendimentos mais organizado e eficiente. A seguir, são apresentadas cada uma das telas que compõem esse módulo.

Tela da Lista de Serviços cadastrados

A tela inicial do módulo de Serviços apresenta uma visão clara e organizada de todos os serviços disponíveis na clínica. Logo no topo, o usuário encontra os filtros do catálogo, que facilitam a busca por itens específicos. É possível pesquisar por nome, filtrar por tipo de serviço (como consultas, exames, outros cuidados) e refinar pela função do profissional associado – médico, enfermeiro ou demais responsabilidades. Esses filtros tornam a navegação extremamente eficiente, especialmente em um catálogo amplo em constante crescimento.

Ainda na mesma seção, a interface oferece duas opções muito úteis: Exportar para Excel, que transforma todo o catálogo filtrado em uma planilha estruturada, e limpar filtros, que reseta facilmente a página. Essa experiência foi pensada para auxiliar administradores e gestores em rotinas de manutenção, auditoria e análise de valores sugeridos.

Logo abaixo, aparece a lista de Serviços cadastrados, onde cada card exibe de forma compacta e objetiva as principais informações: nome do serviço, tipo, profissional indicado, descrição do que está incluído e o valor sugerido – sempre acompanhado da data da última atualização. Essa combinação garante rastreabilidade e transparência sobre mudanças realizadas ao longo do tempo. Cada

item conta com um botão “Editar”, que conduz rapidamente ao formulário completo do serviço selecionado.

Serviços

Cadastre os serviços que serão utilizados nos atendimentos, incluindo valores sugeridos e profissional/função.

Filtros do catálogo

Busque por nome, tipo ou profissional/função.

Buscar por nome: Digite parte do nome

Tipo de serviço: Todos os tipos

Profissional ou função: Todos os profissionais

Exportar para Excel | Limpar filtros

Serviços cadastrados

Use estes serviços ao registrar novos atendimentos.

Serviço	Valor sugerido	Última atualização	Ação
Acompanhamento pós-operatório (consulta)	R\$ 160,00	23/11/2025	<input type="button"/> Editar
Acompanhamento pós-operatório (enfermagem)	R\$ 120,00	23/11/2025	<input type="button"/> Editar
Aplicação de medicação injetável	R\$ 70,00	23/11/2025	<input type="button"/> Editar

Figura 85. Lista de serviços

Tela de Novo Serviço

Ao escolher a opção "Novo serviço", o usuário é levado para um formulário claro e direto, projetado para agilizar o processo de inclusão de novos itens no catálogo. A primeira etapa do formulário solicita o nome do serviço, descrevendo de forma objetiva sua finalidade. Em seguida, o usuário escolhe o tipo de serviço, que organiza o catálogo entre categorias como consultas, exames ou outros cuidados. Isso ajuda a manter uma padronização importante para o fluxo de atendimentos.

O campo de profissional indicado é essencial para garantir que o serviço seja corretamente filtrado no momento da criação de um atendimento. Assim, veterinários, enfermeiros e demais profissionais aparecem associados às suas responsabilidades típicas. Já no campo Valor sugerido, o usuário informa um valor de referência que será automaticamente aplicado quando o serviço for selecionado nos atendimentos – sempre podendo ser ajustado conforme necessidade.

Figura 86. Tela de cadastro de serviço

Por fim, o campo de Descrição permite registrar detalhes do procedimento, como instrumentos utilizados, recomendações, escopo do atendimento e outras observações relevantes. A intenção é garantir que cada serviço seja autossuficiente em informações para orientar o processo clínico. O botão “Salvar serviço” conclui o cadastro, enquanto a opção “Cancelar” mantém o fluxo seguro caso o usuário desista da operação.

Tela de Editar Serviço

A tela de edição de serviços apresenta a mesma estrutura visual do formulário de criação, mas já preenchida com os dados existentes. Isso facilita atualizações rápidas, correções de informação ou ajustes de valores sugeridos. O usuário encontra imediatamente o nome do serviço, que pode ser adaptado caso haja mudanças no escopo da clínica.

Em seguida, o tipo de serviço pode ser alterado, permitindo reorganizar o catálogo conforme novas diretrizes internas. O campo de profissional indicado também pode ser ajustado – especialmente útil em contextos em que equipes crescem e responsabilidades são redistribuídas.

Editar serviço

Cadastre serviços para agilizar futuros atendimentos.

Nome do serviço	Tipo de serviço
Acompanhamento pós-operatório (enfermagem)	Outros cuidados
Profissional indicado	Valor sugerido
Enfermeiro	120.00
Descrição	
Cuidados de enfermagem no pós-operatório, incluindo curati	

Cancelar **Atualizar serviço**

Figura 87. Tela de edição de serviços

O valor sugerido é um dos campos mais atualizados no dia a dia, e a interface dá total autonomia para que ajustes sejam realizados sempre que houver mudanças na tabela de preços. A descrição também pode ser expandida ou revisada conforme a prática evolui, mantendo o catálogo coerente e atualizado.

Depois das revisões, o botão “Atualizar serviço” salva todas as alterações, registrando automaticamente a nova data de atualização exibida na listagem geral.

Cards de serviços para visualização rápida

Cada card exibido na lista de serviços possui um papel importante: oferecer uma visualização rápida e funcional dos itens disponíveis. Nele, o usuário vê imediatamente o nome do serviço com destaque, seguido pelo tipo e pelo profissional indicado. A descrição complementar resume o que está incluído no procedimento, ajudando equipes administrativas e clínicas a compreenderem detalhadamente o escopo de cada item.

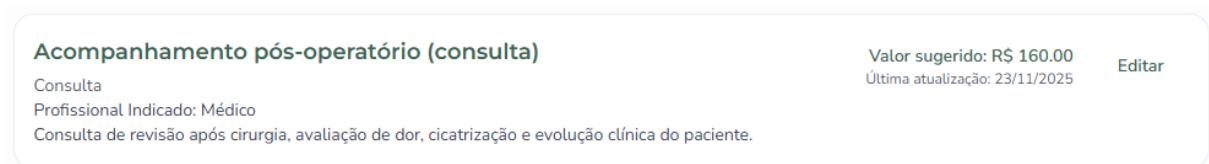


Figura 88. Card de serviço

À direita, o valor sugerido aparece em destaque, reforçado pela última data de atualização. Esse conjunto de informações facilita consultas rápidas, comparações entre serviços e decisões baseadas em dados atualizados.

O botão Editar permanece disponível em cada card, garantindo agilidade no processo de manutenção do catálogo.

2.11.8. Produtos

A seção Produtos reúne todo o catálogo de itens utilizados na rotina da clínica – desde medicamentos até insumos gerais. É aqui que a Aurauet centraliza o controle de estoque, custos, preços de venda e disponibilidade comercial, garantindo que a equipe tenha sempre informações atualizadas para tomar decisões rápidas e evitar rupturas.

O módulo foi projetado para oferecer uma visão clara, organizada e dinâmica, permitindo filtros avançados, exportação em Excel e acesso imediato a ações críticas como editar, remover e ajustar estoque.

Tela Principal - Catálogo e gestão de prateleira

A tela inicial do módulo apresenta, logo no topo, um conjunto de filtros inteligentes que facilitam a busca por qualquer item do catálogo. O usuário pode pesquisar por parte do nome, filtrar pelo status (ativo, inativo) e pela disponibilidade (disponível para venda ou indisponível). Essa filtragem instantânea acelera o acesso a produtos específicos e já prepara o terreno para quem precisa exportar dados diretamente para Excel.

Logo abaixo, a seção Controle de prateleira oferece uma visão consolidada dos produtos cadastrados. Cada card traz informações essenciais para tomada de decisão: descrição resumida, custo, preço de venda, estoque atual e o estoque mínimo recomendado – além de dois indicadores importantes: o status do produto e sua disponibilidade para venda.

The screenshot shows a web-based application for managing a product catalog. At the top, there's a header with the word "Produtos" and a "Novo produto" button. Below the header, a section titled "Filtros do catálogo" allows users to search by name, filter by status (Todos), and filter by availability (Todas). There are buttons to "Exportar para Excel" and "Limpar filtros". A section titled "Controle de prateleira" displays 20 products. The first product listed is "Analgésico Injetável", which is described as an injectable analgesic for dogs and cats. It shows a cost of R\$ 28.00 and a selling price of R\$ 78.00. The current stock is 9 units, with a minimum recommended of 3. It has an "ATIVO" status and is marked as "INDISPONÍVEL PARA VENDA" (Not available for sale). To the right of the product card are three actions: "Editar", "Ajustar estoque", and "Remover". The second product listed is "Anti-inflamatório Oral Canino 20mg", described as a non-steroidal anti-inflammatory drug for dogs, used under veterinary prescription. It costs R\$ 22.00 and sells for R\$ 59.90. The current stock is 18 units, with a minimum recommended of 5. It has an "ATIVO" status and is marked as "DISPONÍVEL PARA VENDA" (Available for sale). To the right of this product are the same three actions: "Editar", "Ajustar estoque", and "Remover".

Figura 89. Tela da lista de Produtos

A navegação fica ainda mais fluida com as ações rápidas ao lado direito: Editar, ajustar estoque e remover. Tudo foi pensado para reduzir cliques e agilizar o trabalho da equipe.

Tela de cadastro de produto (Novo Produto)

Ao clicar em Novo produto, o usuário é levado a uma tela simples, clara e direta, onde pode registrar todos os detalhes necessários para manter o catálogo completo. O formulário permite informar nome, descrição, custo, preço de venda, estoque atual, estoque mínimo, status e disponibilidade.

A interface foi cuidadosamente desenhada para orientar o preenchimento passo a passo, sem poluição visual, garantindo que o cadastro flua de forma natural mesmo para usuários iniciantes. A ação cadastrar finaliza o processo com um retorno visual imediato de sucesso.

Novo produto

Preencha os dados financeiros e de estoque para manter o cuidado completo em cada detalhe.

Nome	<input type="text" value="Nome do produto"/>		
Descrição	<input type="text" value="Descrição opcional"/>		
Custo	<input type="text" value="0"/>	Preço de venda	<input type="text" value="0"/>
Estoque atual	<input type="text" value="0"/>	Estoque mínimo	<input type="text" value="0"/>
Status	<input type="text" value="Ativo"/>	Disponibilidade	<input type="text" value="Disponível para venda"/>

[Cancelar](#) **Cadastrar**

Figura 90. Cadastro de novos produtos

Tela de edição de produto

Editar produto

Preencha os dados financeiros e de estoque para manter o cuidado completo em cada detalhe.

Nome	<input type="text" value="Analgésico Injetável"/>		
Descrição	<input type="text" value="Analgésico de uso injetável para controle de dor aguda em cães e gatos."/>		
Custo	<input type="text" value="28"/>	Preço de venda	<input type="text" value="78"/>
Estoque atual	<input type="text" value="9"/>	Estoque mínimo	<input type="text" value="3"/>
Status	<input type="text" value="Ativo"/>	Disponibilidade	<input type="text" value="Indisponível para venda"/>

[Cancelar](#) **Salvar alterações**

Figura 91. Tela para edição de produto

Quando um item do catálogo precisa ser alterado – seja uma atualização de preço, ajuste de descrição ou mudança de disponibilidade – a tela de edição entra em cena. Ela é praticamente idêntica ao formulário de cadastro, mas já carregando todas as informações atuais do produto.

Essa familiaridade visual reduz curva de aprendizado e facilita tanto pequenas correções quanto atualizações mais complexas, sempre com foco em precisão e confiabilidade no registro dos dados.

Tela de ajuste de estoque



Figura 92. Tela para ajuste de estoque

A gestão de estoque exige decisões rápidas, e por isso a Auravet trata o ajuste como uma ação própria, com sua própria tela. Nela, o usuário simplesmente informa a quantidade a ser ajustada – podendo ser positiva para entradas ou negativa para baixas.

O sistema reforça esse comportamento com um exemplo diretamente na interface ("5 para entrada, -2 para baixa"), tornando o preenchimento intuitivo e evitando erros. Após confirmar, o estoque é atualizado instantaneamente, refletindo o movimento no catálogo principal.

Card de Produto



Figura 93. Card de produto

Por fim, cada produto possui um card resumido, exibido tanto no catálogo quanto após atualizações. Esse card reúne todas as informações essenciais:

- nome e descrição;
- custo e preço de venda;
- estoque atual e mínimo recomendado;
- status e disponibilidade;
- ações rápidas de edição, ajuste e remoção.

Essa visão consolidada permite que o usuário consulte o estado do produto em poucos segundos e tome decisões imediatas, sem necessidade de abrir telas adicionais.

2.11.9. Atendimento

O módulo de Atendimento concentra o coração operacional da Auravet. É aqui que a equipe clínica registra cada consulta, procedimento, medicação aplicada ou serviço realizado para um pet – sempre de forma clara, organizada e totalmente integrada aos demais módulos do sistema.

O atendimento pode ser iniciado a partir de vários pontos da plataforma: diretamente pela agenda, pelo histórico do pet, pela lista geral de atendimentos concluídos, ou até mesmo pela própria tela de agendamentos confirmados. Em todos os casos, os dados chegam pré-preenchidos para acelerar o fluxo da equipe.

A seguir, a descrição de cada parte da tela e do fluxo.

Lista de Atendimentos

A tela de Atendimentos funciona como um painel central de acompanhamento de tudo o que está acontecendo na clínica, em tempo real. Logo no topo, um conjunto de filtros permite refinar a visualização por status do atendimento, período, tutor e pet, tornando simples localizar desde um caso específico até um conjunto de atendimentos de um determinado dia ou profissional. Acima da lista principal, os botões “Exportar para Excel” e “Registrar atendimento” dão suporte tanto às necessidades administrativas quanto à operação clínica, permitindo gerar relatórios estruturados e abrir novos registros diretamente, sem depender da agenda.

Atendimentos

Visualize, filtre e gerencie todos os atendimentos da clínica em um só lugar.

Filtros

Refine a lista por status, período, tutor ou pet.

Status De Até

Tutor Pet

Limpar filtros

Lista de atendimentos

Ações rápidas para cada atendimento.

Outros cuidados EM ANDAMENTO
29/11/2025, 12:00:00
R\$ 239,90
Pet: Ônix • Tutor: Nair Adriana Caldeira
RESPONSÁVEL: DR. HENRIQUE SALES PACHECO

Outros cuidados EM ANDAMENTO
28/11/2025, 12:00:00
R\$ 585,00
Pet: Corvo • Tutor: Calebe Julio Rodrigues
RESPONSÁVEL: DRA. CAMILA TEIXEIRA LINS

Figura 94. Tela completa de Atendimentos – filtros e lista.

Abaixo dos filtros, a lista de atendimentos é exibida em formato de cartões empilhados, cada um trazendo um resumo fiel da consulta: título do atendimento, data e horário, valor total, identificação do pet e do tutor, além do profissional responsável. O selo de status (“Em andamento”, “Concluído”, “Cancelado”) facilita a leitura rápida, ajudando a equipe a enxergar quais casos ainda precisam de alguma ação. Essa combinação entre filtros, lista e botões rápidos transforma a tela em um verdadeiro painel de controle da rotina clínica.

Card de atendimento e botões de ação

Cada atendimento é representado por um cartão visualmente limpo, que concentra todas as informações críticas em poucos elementos. No topo, o nome do atendimento – por exemplo, “Outros cuidados” – é acompanhado de um selo de status em destaque, indicando imediatamente em que etapa aquele caso se

encontra. Logo abaixo, são apresentados data e hora, valor total, identificação do pet e do tutor, além do nome do profissional responsável. Essa estrutura foi pensada para permitir que o colaborador, com uma única leitura, entenda “quem, quando, quanto e com quem” está relacionado àquele atendimento.

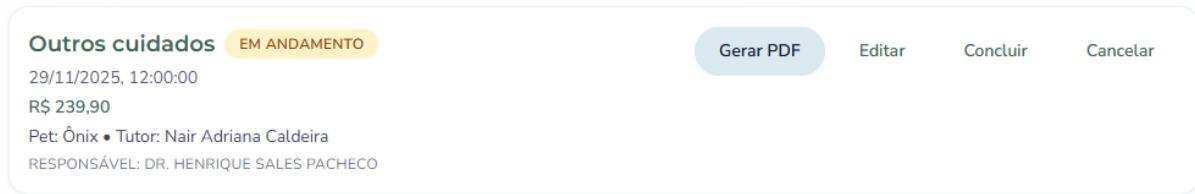


Figura 95. Card de atendimento em destaque – Outros cuidados.

Na lateral direita do cartão ficam os botões de ação, que traduzem os principais caminhos possíveis para aquele caso. O botão “Gerar PDF” permite criar rapidamente um documento para impressão ou envio ao tutor, contendo os dados essenciais do atendimento. As opções “Editar”, “Concluir” e “Cancelar” representam as decisões operacionais do dia a dia: ajustar informações clínicas ou financeiras, finalizar o atendimento quando todo o registro já está completo ou cancelar um registro que não deve seguir adiante. Dessa forma, o cartão não é apenas um resumo visual, mas um ponto de partida para todas as ações relacionadas àquele atendimento.

Dados do atendimento

A primeira seção da tela apresenta todas as informações essenciais para identificar o atendimento. Se o usuário tiver vindo a partir de um agendamento, esse compromisso já aparece selecionado automaticamente, incluindo tutor, pet, profissional responsável e horário.

Caso o atendimento não tenha origem em um agendamento, o usuário pode simplesmente escolher o pet e o tutor manualmente, além de informar o horário de início e, se desejar, o horário de término – que também pode ser definido somente ao final para registrar a duração real.

Registrar Atendimento

Combine múltiplos serviços do catálogo e produtos utilizados para gerar um atendimento completo.

Dados do atendimento

Preencha com atenção para manter o histórico impecável e inclua quantos serviços forem necessários no mesmo atendimento.

Agendamento (opcional)

25/11/2025 10:00 - 11:00 • Ônix (Nair Adriana Caldeira)

Pet	Tutor
Ônix — Tutor(a): Nair Adriana Caldeira	Nair Adriana Caldeira

Tutor definido automaticamente a partir do pet selecionado.

Início do atendimento	Término do atendimento
25/11/2025 10:00	25/11/2025 11:00

Opcional: preencha ao concluir para registrar a duração.

Profissional responsável	Profissional assistente
Dra. Bianca Correia Menezes — bianca.menezes@auravet.com.br	Enf. Gustavo Vieira Campos — gustavo.campos@auravet.com.br

Figura 96. Registro de atendimento – Dados do Atendimento

Essa etapa centraliza quem é o animal, quem o acompanha, qual profissional executa, quem auxilia e quando o atendimento ocorreu, garantindo padronização e rastreabilidade das informações clínicas.

Registro dos serviços

Logo abaixo, o profissional pode registrar todos os serviços realizados durante o atendimento.

Cada serviço permite escolher um item do catálogo já configurado na Auravet, trazendo automaticamente o preço sugerido, a descrição e demais orientações.

O usuário pode ajustar a quantidade, sobreescrivver valores se necessário e adicionar observações clínicas específicas, enriquecendo o prontuário com detalhes importantes.

Serviço	Quantidade	Preço sugerido (R\$)
Monitorização de paciente internado (po)	1	190,00

Selecionar um serviço do catálogo

Observações

Detalhes relevantes sobre este serviço

Figura 97. Registro de serviço no atendimento

A cada item registrado, subtotais são calculados automaticamente, e um resumo consolidado dos serviços aparece logo abaixo, ajudando na conferência rápida.

Registro de Produtos

Na sequência, o sistema permite registrar produtos e insumos utilizados no atendimento – como medicamentos, materiais de curativo, suplementos e outros itens integrados ao estoque.

Ao selecionar um produto, o sistema informa disponibilidade, estoque mínimo e atualiza automaticamente o estoque pós-uso, garantindo controle preciso e evitando rupturas.

Assim como nos serviços, é possível ajustar quantidade, preço unitário e observar detalhes relevantes.

Produto	Quantidade	Preço unitário (R\$)
Antibiótico de Amplo Espectro Injetável	3	92,00

Selecionar um produto

Subtotal: R\$ 0,00

Remover

Resumo dos produtos

Selecionar um produto para o item 1.

Total dos produtos: R\$ 0,00

Figura 98. Registro de produto no atendimento

O resumo da seção apresenta todos os produtos usados e o valor total correspondente.

Registros de prontuário de atendimento

Para complementar o registro, o profissional pode inserir anotações livres sobre o atendimento – desde percepções clínicas até orientações repassadas ao tutor ou comportamento do pet durante o procedimento.

Esse campo é fundamental para manter um prontuário completo, estruturado e com continuidade clínica para atendimentos futuros.

Prontuário do atendimento

Nova entrada

Descreva evolução clínica, condutas e orientações.

Entradas ficam registradas com autor e horário.

Registrar entrada

Nenhum registro clínico ainda. Adicione a primeira entrada para montar o prontuário.

Figura 99. Registro de prontuário de atendimento

Totais e Ações finais

Ao final da página, o sistema apresenta o valor total do atendimento, somando automaticamente serviços e produtos.

Nesse mesmo bloco, o usuário encontra as ações finais:

- Gerar PDF do atendimento, criando um documento profissional com todos os registros, ideal para impressão ou entrega ao tutor.
- Limpar atendimento, caso deseje reiniciar o preenchimento.
- Salvar atendimento, concluindo o processo e registrando tudo no histórico do pet, no relatório financeiro e no painel de produtividade da equipe.

Enf. Paula Regina Saldanha
Ônix precisou de ajuda.

23/11/2025, 14:56:00

Totais do atendimento

Serviços: R\$ 140.00 Produtos: R\$ 99.90 Total geral: R\$ 239.90

Gerar PDF do atendimento Limpar Salvar atendimento

Figura 100. Botões do Registro de Atendimentos

Botões de ação no atendimento

Na tela interna do atendimento, os botões de ação posicionados ao final do formulário foram desenhados para refletir exatamente as possibilidades do fluxo clínico. Ao longo da tela, o profissional registra anamnese, serviços realizados, produtos utilizados e observações clínicas. Quando chega à seção de totais, os campos de resumo (serviços, produtos e valor geral) ajudam a conferir se tudo está

coerente antes de tomar a decisão final. É nesse contexto que surgem os botões “Cancelar atendimento”, “Concluir atendimento”, “Gerar PDF do atendimento”, “Limpar” e “Salvar atendimento”.



Figura 101. Rodapé da tela de atendimento – botões de ação.

O botão “Salvar atendimento” permite registrar o estado atual como “Em andamento”, ideal para situações em que o caso ainda está em progresso ou quando o profissional precisa registrar informações por etapas. “Concluir atendimento” é utilizado quando todos os dados já estão revisados, gerando ou atualizando automaticamente a fatura correspondente no módulo de Caixa e alterando o status para concluído.

“Gerar PDF do atendimento” produz um documento formal com o resumo clínico e financeiro, pronto para ser compartilhado. Já “Cancelar atendimento” marca aquele registro como cancelado, preservando o histórico, mas impedindo sua continuidade como caso ativo. Por fim, o botão “Limpar” atua como um recurso de segurança para descartar alterações ainda não salvas, sem comprometer os dados já persistidos anteriormente. Essa combinação de ações oferece controle total ao profissional, sem perder a simplicidade da interface.

2.11.10. Condições de pagamento

O módulo de Formas e condições de pagamento traduz a política financeira da clínica em elementos claros e reutilizáveis, garantindo que o time de caixa e financeiro trabalhe sempre com as mesmas regras.

Lista de condições de pagamento

o acessar essa área, o usuário encontra uma tabela organizada com todas as condições cadastradas, exibindo nome, prazo em dias, número de parcelas e observações. Condições como “À vista”, “30 dias”, “60 dias” ou “Cartão 3x”

aparecem em uma lista uniforme, tornando evidente quais opções podem ser aplicadas na hora de finalizar uma cobrança.

Formas e condições				
Condições cadastradas				
Edite ou remova condições de pagamento utilizadas pelo caixa.				
NOME	PRAZO (DIAS)	PARCELAS	OBSERVAÇÕES	AÇÕES
30 dias	30	1	Vencimento em 30 dias	Editar Remover
60 dias	60	1	Vencimento em 60 dias	Editar Remover
À vista	1	1	Pagamento imediato	Editar Remover
Cartão 2x	30	2	Parcelado em 2x com intervalos de 30 dias	Editar Remover
Cartão 3x	30	3	Parcelado em 3x com intervalos de 30 dias	Editar Remover
Cartão 4x	30	4	Parcelado em 4x com intervalos de 30 dias	Editar Remover

Figura 102. Lista de condições de pagamento cadastradas.

Cada linha dessa tabela possui ações de Editar e Remover, permitindo que a clínica ajuste sua política de cobrança conforme evoluí o modelo de negócio.

Nova condição ou editar

Ao clicar em Nova condição, o sistema abre um formulário dedicado, no qual o usuário define o nome da condição, o prazo, a quantidade de parcelas e uma descrição complementar que orienta o time na prática – por exemplo, “Parcelado em 3x com intervalos de 30 dias” ou “Pagamento imediato no ato do atendimento”.

Nova condição

Defina prazos, parcelas e uma descrição clara para orientar o time do caixa.

Nome da condição	Ex.: Cartão de crédito, Boleto em 30 dias		
Prazo em dias	0	Número de parcelas	1
Observações	Instruções adicionais, taxas ou combinados específicos		

[Cancelar](#) [Salvar condição](#)

Figura 103. Formulário Nova condição

Esse mesmo formulário é reutilizado no modo de edição, exibindo os campos já preenchidos para ajustes pontuais.

Figura 104. Formulário Editar condição

Com isso, sempre que o colaborador estiver no fluxo de Caixa ou Financeiro, poderá escolher uma condição de pagamento padronizada, em vez de criar combinações soltas de prazos e parcelas. O resultado é um fluxo de cobrança mais claro, menos sujeito a erros e perfeitamente alinhado à proposta da Auravet de unir organização administrativa e cuidado humano em cada etapa do atendimento.

2.11.11. Caixa

O módulo de Caixa é onde toda a jornada clínica da Auravet se transforma em números organizados, previsibilidade e segurança financeira. Depois que o pet passa por atendimento, os serviços e produtos utilizados geram uma fatura, que por sua vez aparece aqui como uma conta a receber, pronta para ser acompanhada do início ao fim. Essa tela foi pensada para que o time enxergue, em poucos segundos, quanto a clínica tem a receber, o que já foi pago e quais contas ainda exigem ação – seja cobrança, renegociação ou apenas registro do pagamento.

O visual segue a identidade suave da Auravet, com cartões em tons de verde-claro e fundo limpo, transmitindo a ideia de transparência e calma mesmo quando o volume de faturas é alto. No topo, o usuário é recebido por um resumo financeiro que funciona como um painel executivo; logo abaixo, filtros permitem recortar a

visão por tutor, status e período; e, na sequência, a lista de contas apresenta cada fatura com todos os seus detalhes. A partir dessa tela, o colaborador consegue abrir os detalhes da conta, ajustar itens, configurar forma e condição de pagamento, registrar parcelas e marcar a fatura como quitada – tudo dentro de um fluxo contínuo e sem ruídos.

The screenshot displays the 'Caixa' module interface. At the top, there's a header with the title 'Caixa' and a subtitle: 'Acompanhe contas a receber, veja detalhes dos itens faturados e registre pagamentos com transparência.' Below this is a 'Resumo financeiro' section with three cards:

- TOTAL EM ABERTO**: R\$ 3.963,70 (10 contas aguardando pagamento)
- TOTAL RECEBIDO**: R\$ 0,00 (Somatório das parcelas marcadas como pagas. Quitadas: R\$ 0,00 • 0 faturas)
- TOTAL QUITADO**: R\$ 0,00 (0 faturas totalmente liquidadas)

Below the summary is a 'Filtrar contas' section with filters for Tutor, Status, Date range (De and Até), and buttons for 'Aplicar filtros', 'Exportar para Excel', and 'Limpar'.

The main area shows a table titled 'Contas a receber' with columns: TUTOR, TOTAL, PAGAMENTO, STATUS, and AÇÕES. One row is visible for 'Victor Henry da Rosa' with details: Total R\$ 188,00, Payment method 'Dinheiro', Status 'Em aberto', and Actions buttons for 'Ver detalhes' and 'Gerar PDF'.

Figura 105. Visão geral do módulo Caixa.

Resumo Financeiro

Logo no topo da tela, o colaborador encontra o Resumo financeiro, composto por três cartões principais: Total em aberto, Total recebido e Total quitado.

Cada card apresenta o valor consolidado e um texto auxiliar explicando o que ele representa na prática, como a quantidade de contas aguardando pagamento ou o somatório das parcelas já marcadas como pagas.

Essa seção funciona como um “termômetro” da clínica: em poucos segundos, o usuário entende se o fluxo de recebimentos está saudável ou se existem pontos de atenção imediata.



Figura 106. Cards do Resumo financeiro.

Filtros de contas

Logo abaixo do resumo, a interface apresenta a faixa de filtros, que organiza o acesso às informações de forma inteligente. Nessa área o usuário pode escolher um tutor específico, filtrar por status da fatura (em aberto, parcialmente paga, quitada) e definir um intervalo de datas utilizando campos com seleção por calendário. A ideia é permitir recortes muito objetivos, como “todas as contas em aberto deste mês”, “tudo o que já foi recebido no período” ou “faturas de um tutor específico”.

Os botões na base – Aplicar filtros, Exportar para Excel e Limpar – estruturam o fluxo de trabalho: o colaborador pode refinar a lista, gerar um relatório em planilha para controles externos ou simplesmente limpar os critérios e voltar à visão geral. Tudo isso acontece em uma área visualmente clara e espaçada, para que a equipe consiga trabalhar com filtros diariamente sem esforço cognitivo desnecessário.

The figure shows a filtering interface with the following fields:

- Tutor:** Todos os tutores
- Status:** Todos os status
- De:** dd/mm/aaaa
- Até:** dd/mm/aaaa
- Buttons:** Aplicar filtros, Exportar para Excel, Limpar

Figura 107. Seção de filtros do Caixa.

Lista de contas a receber

A seguir, a seção de Contas a receber apresenta, em formato de tabela, cada fatura emitida para os tutores. Apesar da estrutura tabular, cada linha se comporta como um cartão rico em informações: na primeira coluna, os dados do tutor aparecem de forma completa, com nome, e-mail, telefone, CPF e endereço; na sequência, o valor total da fatura é exibido com destaque, seguido pelos dados de pagamento e pelo status da cobrança.

A coluna de pagamento traz a forma utilizada (como dinheiro, cartão ou outra modalidade) e a condição aplicada, por exemplo “Cartão 2x” ou “30 dias”. Já o status é representado por uma etiqueta em tom suave, que indica se aquela conta está em aberto, parcialmente quitada ou totalmente paga. Na extremidade direita, os links “Ver detalhes” e “Gerar PDF” funcionam como atalhos para o dia a dia: o primeiro abre a visão aprofundada da fatura, enquanto o segundo gera o documento em PDF, pronto para ser salvo, impresso ou enviado ao tutor.

Essa lista forma o “backbone” do módulo Caixa, centralizando todas as contas em um só lugar e permitindo que a equipe navegue de forma rápida entre consulta, análise e ação.

Contas a receber				
Detalhes completos das contas emitidas para cada tutor.				
TUTOR	TOTAL	PAGAMENTO	STATUS	AÇÕES
Victor Henry da Rosa victor.henry.darosa@live.ie 14994781561 CPF: 959.751.988-79 Rua Anelda Volta Brazini, 343 • Antonio Carlos Nascimento da Silva • Marília - SP • CEP 17523-893	R\$ 188,00	Cartão de crédito Cartão 2x • Parcelado em 2x com intervalos de 30 dias 0/2 parcelas pagas Próx. 30/12/2025	Em aberto	Ver detalhes Gerar PDF
Nair Adriana Caldeira nair.caldeira@focusdm.com.br 14997774000 CPF: 199.513.588-70 Rua Carolina Moraes Almeida, 545 • Senador Salgado Filho • Marília - SP • CEP 17502-290	R\$ 239,90	Dinheiro À vista • Pagamento imediato 0/1 parcelas pagas Próx. 05/12/2025	Em aberto	Ver detalhes Gerar PDF

Figura 108. Linha de conta a receber na lista do Caixa.

Detalhes da conta

Ao clicar em “Ver detalhes”, o sistema abre um painel dedicado àquela cobrança, pensado para oferecer uma visão completa e ao mesmo tempo organizada. No topo, aparecem os dados do tutor e o status da fatura, além do valor total, da forma de pagamento, da condição escolhida, da situação das parcelas e do responsável interno pelo registro no caixa. À direita, um link permite baixar o PDF da fatura diretamente dessa tela.

Logo abaixo, a área de Itens faturados apresenta os serviços e produtos que compõem a conta. Cada linha identifica o tipo de item, a quantidade, o valor unitário, o valor total e, no caso de serviços, a data em que foram realizados e o pet atendido. Essa estrutura garante rastreabilidade entre o atendimento clínico, o uso de insumos e a cobrança ao tutor, fortalecendo tanto o controle interno quanto a transparência no relacionamento com o cliente.

Figura 109. Detalhes da conta.

Pagamentos e parcelas

Dentro da mesma tela, a seção de Pagamento assume o papel de “cérebro financeiro” da fatura. Nela o colaborador escolhe a forma de pagamento, como dinheiro, cartão de crédito ou outra modalidade, e seleciona a condição de pagamento cadastrada anteriormente, por exemplo “À vista”, “30 dias”, “Cartão 2x” ou “Cartão 3x”. A partir dessas escolhas, o sistema distribui automaticamente o valor total em parcelas, que passam a ser exibidas em sequência.

Cada parcela é apresentada com campos para valor, data de vencimento e data em que foi paga, além de um checkbox que marca quando o pagamento daquela parcela foi efetivamente recebido. No cabeçalho do bloco, uma pequena linha informa o total das parcelas, ajudando a conferir se a soma continua alinhada com o valor global da fatura. Esse desenho garante que o registro de recebimentos seja feito com precisão, mesmo em situações de pagamentos fracionados, renegociações ou ajustes de datas.

The screenshot shows a user interface for managing payments and installments. At the top, it says "PAGAMENTO" and "Defina a forma de pagamento e ajuste os vencimentos das parcelas desta conta." To the right, it displays "Total das parcelas: R\$ 188,00". Below this, there are two sections for "Parcela 1" and "Parcela 2". Each section has dropdown menus for "Forma de pagamento" (set to "Cartão de crédito") and "Condição" (set to "Cartão 2x (2x • 30 dias) • Parcelado em 2x com inter"). Each section also includes fields for "Valor (R\$)" (94,00), "Vencimento" (31/12/2025), "Pago em" (dd/mm/aaaa), and a checkbox labeled "Pago".

Figura 110. Seção de pagamento e parcelas.

Itens extras

Pensando em rotinas reais de clínica – como venda de produtos de balcão, complementos de tratamento ou cobranças avulsas – a tela de detalhes da conta inclui a área de Itens extras. Quando nenhum adicional foi inserido, o usuário vê

apenas uma pequena caixa com um texto explicativo e o botão “Adicionar item”, deixando claro que a fatura aceita incrementos posteriores ao atendimento inicial.

Ao abrir o formulário, o sistema oferece duas abordagens: por padrão, a opção “Produto em estoque” fica ativa, permitindo selecionar um item já cadastrado no catálogo, informar uma descrição mais específica (como “Coleira antipulgas – tamanho M”), definir quantidade e valor unitário. O total estimado é calculado automaticamente, evitando erros simples de digitação. Já a aba “Descrição livre” atende casos em que a clínica precisa registrar uma cobrança não vinculada ao estoque, como uma taxa administrativa ou serviço eventual. Em ambos os modos, o botão “Adicionar item à conta” conclui a operação, atualizando imediatamente o total da fatura.



Figura 111. Botão para registrar itens extras.

Essa flexibilidade faz com que o módulo Caixa acompanhe a dinâmica real do consultório, em que nem tudo nasce dentro do fluxo padrão de atendimento, mas ainda assim precisa ser registrado de forma organizada.

The image shows an open form for managing extra items. At the top, it says "Itens extras" and "Registre produtos vendidos no balcão ou cobranças avulsas para esta conta." To the right is a blue button "Ocultar formulário". Below this are two tabs: "Produto em estoque" (selected) and "Descrição livre". The "Produto em estoque" tab includes a dropdown menu labeled "Selecionar o produto em estoque". The "Descrição livre" tab has a text input field with the placeholder "Ex.: Produto avulso: Coleira antipulgas". There are two input fields for "Quantidade" (containing "1") and "Valor unitário (R\$)". At the bottom, a summary line says "Total estimado: R\$ 0,00" and a green button "Adicionar item à conta" is visible.

Figura 112. Formulário de Itens extras aberto.

Ações finais do Caixa

Na base da tela de detalhes, três botões encerram o fluxo de trabalho com clareza: “Fechar” permite apenas sair da visualização, quando o objetivo do usuário era consultar informações; “Salvar fatura” registra qualquer alteração feita na conta – como inclusão de itens, mudança de forma de pagamento ou ajuste em datas de vencimento – sem necessariamente marcar a fatura como recebida; e “Registrar pagamento” é a ação que consolida o recebimento parcial ou total, atualizando o status da conta e refletindo essa mudança no Resumo financeiro do topo.

Forma de pagamento		Condição	
Cartão de crédito		Cartão 2x (2x • 30 dias) • Parcelado em 2x com intervalo	
Parcela 1 <div style="display: flex; justify-content: space-between;"> <div>Valor (R\$)</div> <div>Vencimento</div> <div>Pago em</div> </div> <div style="display: flex; justify-content: space-around;"> <input type="text" value="94,00"/> <input type="text" value="31/12/2025"/> <input type="text" value="dd/mm/aaaa"/> </div> Parcela 2 <div style="display: flex; justify-content: space-between;"> <div>Valor (R\$)</div> <div>Vencimento</div> <div>Pago em</div> </div> <div style="display: flex; justify-content: space-around;"> <input type="text" value="94,00"/> <input type="text" value="30/01/2026"/> <input type="text" value="dd/mm/aaaa"/> </div>			
<input type="button" value="Fechar"/> <input type="button" value="Salvar fatura"/> <input style="background-color: #2e6b2e; color: white; border-radius: 10px; padding: 5px 10px; border: none; font-weight: bold; margin-left: 10px;" type="button" value="Registrar pagamento"/>			

Figura 113. Botões para manutenção do caixa.

Com essa combinação de visão consolidada, filtros eficientes, detalhes completos e ações bem separadas, o módulo de Caixa da Auravet garante que a etapa financeira acompanhe o mesmo nível de cuidado aplicado à rotina clínica. O resultado é um fluxo em que a equipe consegue cuidar dos números com a mesma atenção que dedica aos tutores e aos pacientes.

2.11.12. Financeiro

O módulo Financeiro da Auravet foi pensado para dar à coordenação da clínica uma visão estratégica das receitas, sem perder a leveza visual e o tom humano que aparecem em todo o sistema. Enquanto o Caixa é o lugar onde o time registra o recebimento efetivo de cada conta, o Financeiro funciona como um “painel de

“controle” das faturas emitidas: o que já entrou, o que está atrasado, o que ainda vai vencer e quais ajustes precisam ser feitos para manter o fluxo de caixa saudável.

Logo na abertura da tela, o usuário encontra um resumo claro da situação atual da clínica, com foco em previsibilidade. A linguagem é simples, direta e acolhedora, evitando termos excessivamente técnicos e priorizando a tomada de decisão rápida. Todo o layout segue a paleta em tons de verde e off-white, com cartões bem espaçados, tipografia suave e destaque sutis em cores mais intensas apenas onde a atenção realmente precisa ser chamada, como em faturas vencidas.

Financeiro

Visualize rapidamente contas emitidas, receitas recebidas e o que ainda está para vencer.

Resumo de receitas

Acompanhe totais recebidos, quitados e em aberto para manter o fluxo de caixa saudável.

TOTAL RECEBIDO R\$ 0,00 Somatório das parcelas pagas, independente do status. Quitado em faturas: R\$ 0,00 (0)	A RECEBER R\$ 3.963,70 10 contas pendentes	PRÓXIMO VENCIMENTO 28/11/2025	ÚLTIMO RECEBIMENTO Ainda sem registros.
---	--	---	---

Filtrar faturas

Combine período e status para refinar a visão.

Bloqueadas Abertas Quitadas

Todas Em dia Vencidas

Status: Todos os status Vencimento: Todas De: dd/mm/aaaa Até: dd/mm/aaaa

Aplicar filtros Limpar

Faturas

Visualize em um único lugar o que está em aberto ou já foi quitado.

Kauê Carlos Eduardo Silva kauê-silva73@advogadostb.com.br 0/1 parcelas	VENCIDO ABERTA	Alice Sophie Simone Nogueira alice-nogueira95@pgpc.com.br 0/1 parcelas	VENCIDO ABERTA
VALOR R\$ 438,70	PRÓXIMO VENCIMENTO 28/11/2025	PAGAMENTO Não definida	VALOR R\$ 494,50
Gerar PDF Ajustar fatura		Gerar PDF Ajustar fatura	

Figura 114. Imagem geral do módulo Financeiro.

Resumo de receitas

Na parte superior da tela, o bloco de Resumo de receitas apresenta, em um só olhar, os principais indicadores financeiros da clínica. Um card em verde suave mostra o Total recebido, somando todas as parcelas já pagas, independentemente do status das faturas. Ao lado, o card A receber indica o valor total ainda pendente, com a quantidade de contas em aberto.

Do outro lado do painel, dois cartões completam a visão: o Próximo vencimento, que aponta a data mais próxima em que uma cobrança deve ser paga, e o Último recebimento, que registra quando foi a última entrada de receita. Esses elementos ajudam a equipe a entender se o fluxo de caixa está ativo, parado ou concentrado em certas datas, facilitando decisões como renegociação, ações de cobrança ou campanhas pontuais.

Visualmente, esse bloco reforça a sensação de controle e estabilidade. Não é apenas um conjunto de números: é uma leitura rápida da saúde financeira da clínica.



Figura 115. Resumo de receitas.

Filtros de Faturas

Logo abaixo do resumo, o módulo apresenta a área de Filtros de faturas, que permite refinar a visão de acordo com a necessidade de quem está analisando os dados. O usuário consegue combinar diferentes critérios, como status da fatura (bloqueadas, abertas, quitadas), situação de vencimento (todas, em dia ou vencidas) e intervalos de datas, escolhendo período inicial e final.

O objetivo dessa seção é permitir que o Financeiro enxergue exatamente o recorte que precisa: por exemplo, apenas faturas vencidas no mês atual, apenas contas em dia para conferência, ou todas as faturas abertas em um determinado intervalo para projeção de receita. Os botões “Aplicar filtros” e “Limpar” mantêm o fluxo ágil, incentivando o uso dos filtros como parte natural da rotina, e não como algo complexo ou distante.

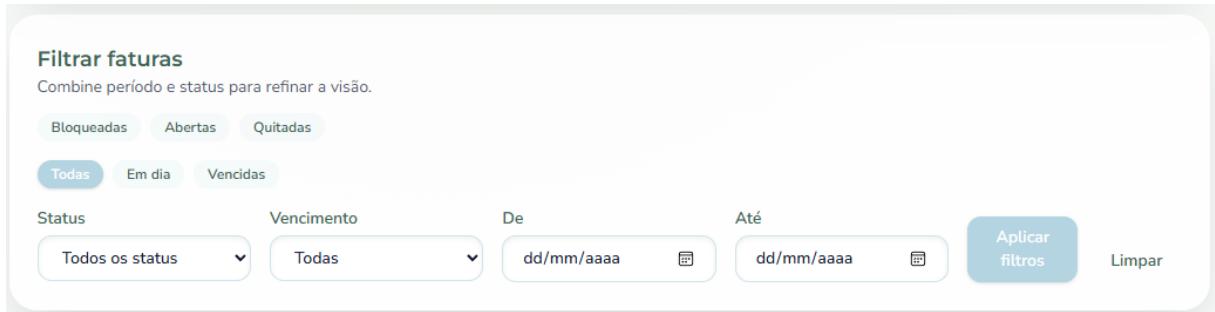


Figura 116. Filtrar faturas.

Lista de Faturas

Abaixo dos filtros, a seção Faturas apresenta cada conta em formato de cartão, organizado por tutor. Cada card mostra, de forma compacta e amigável, o nome do responsável, e-mail de contato e a relação de parcelas daquela fatura. Logo em destaque, o valor total, a data do próximo vencimento e o status de pagamento compõem um pequeno “raio-x” daquela receita.

Etiquetas coloridas indicam rapidamente se a fatura está vencida, aberta ou já teve todas as parcelas quitadas. Esse uso de “chips” visuais ajuda a bater o olho e entender prioridades, sem precisar abrir detalhes uma a uma. Na parte inferior do card, ações diretas permitem gerar PDF da fatura e ajustar fatura, caso seja necessário renegociar prazos, condições ou juros.

Essa combinação entre informação sintetizada e ações rápidas transforma a lista de faturas em um painel operacional eficiente, ideal para fechamento de mês, acompanhamento diário ou reuniões de resultado.

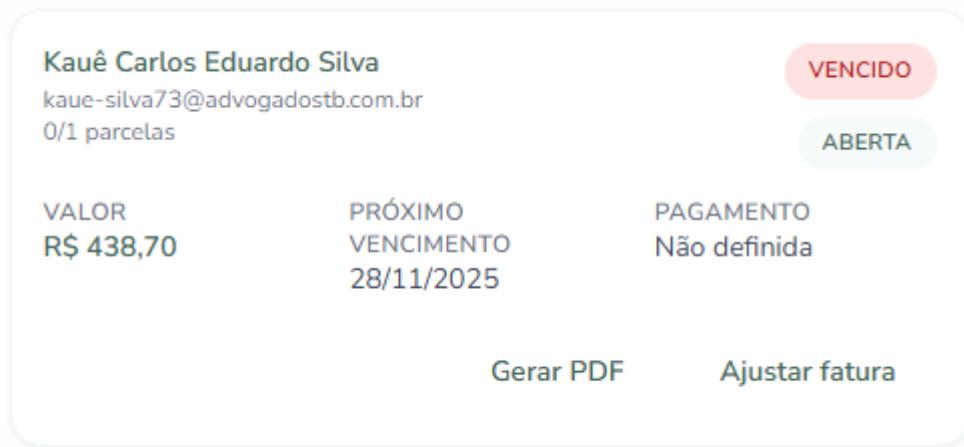


Figura 117. Card de fatura.

Ajuste de fatura e renegociação

Quando o usuário escolhe ajustar fatura, um modal dedicado é aberto para tratar da renegociação com cuidado e transparência. Nessa tela, o colaborador visualiza e edita os principais campos responsáveis pelo cálculo da cobrança: data de vencimento, forma de pagamento, condição de pagamento e percentual de juros aplicado.

À medida que esses campos são ajustados, o sistema recalcula automaticamente o total ajustado e exibe as parcelas recalculadas, indicando valor, data de vencimento e quantidade de parcelas resultantes. A interface foi desenhada para

apoiar o diálogo com o tutor: o profissional consegue simular cenários, definir uma nova condição, incluir ou não juros e, ao final, salvar os ajustes com segurança.

Tudo é apresentado em linguagem clara, com espaçamento confortável entre campos e botões bem destacados para salvar ajustes ou cancelar a operação. Assim, o módulo Financeiro não apenas controla o que entra e o que sai, mas também oferece uma ferramenta prática para negociação e reestruturação de dívidas, reforçando a postura de parceria da Auravet com as famílias dos pacientes.

2.11.13. Usuários

O módulo de Colaboradores é o espaço central para administrar quem faz parte da equipe interna da Auravet. Ele garante controle de acesso, organização dos papéis clínicos e administrativos, além de permitir que a gestão acompanhe disponibilidade e perfis profissionais com precisão. Tudo foi pensado para manter a operação segura, clara e alinhada às responsabilidades de cada membro da equipe.

Lista de colaboradores da Auravet

A tela inicial apresenta todos os usuários ativos no ecossistema Auravet. Cada cartão exibe o nome do colaborador, seu e-mail institucional, função, último acesso e uma visualização rápida do perfil clínico (quando aplicável).

Figura 118. Lista de colaboradores

Os botões Editar e Desativar permitem administrar facilmente a presença e o acesso daquele colaborador no sistema. Desativar remove o acesso sem apagar o histórico – preservando segurança e dados operacionais.

A estrutura foi pensada para que a equipe consiga identificar rapidamente quem é quem, qual o papel de cada pessoa e se ela está disponível na rotina clínica.

Cadastrar novo colaborador

A tela de criação de usuário é simples e completa. Aqui é possível definir:

- Nome completo – registro principal do colaborador.
- E-mail institucional – usado para login; mantém o padrão @aurauet.com.br.
- Senha provisória – enviada para o usuário iniciar o acesso.

- **Função – Contador, Médico, Enfermeiro, Assistente administrativo etc.** É o papel que orienta a atuação no sistema.

Novo colaborador Auravet

Cadastre um novo membro da equipe com papel e senha provisória.

Nome completo	E-mail institucional
Nome e sobrenome	admin@auravet.com
Senha provisória	Função
.....	Contador
Perfil clínico	
Preencha os dados apresentados nos formulários de agendamento e atendimento.	
Especialidade clínica	CRMV
Ex.: Clínica geral, felinos, cirurgia	Registro profissional
Opcional	Opcional
Bio clínica	
Resumo sobre a atuação profissional do colaborador	
Opcional	
Turnos disponíveis	
Selecione os períodos em que o colaborador pode ser escalado.	
<input type="checkbox"/> Manhã <input type="checkbox"/> Tarde <input type="checkbox"/> Noite Deixe todos desmarcados para uma agenda flexível.	

Cancelar **Criar colaborador**

Figura 119. Cadastrar novo colaborador

Além disso, existe toda a seção de Perfil Clínico, essencial para profissionais de saúde:

- **Especialidade clínica**
- **CRMV**
- **Biografia clínica**

Esses dados enriquecem as telas de Agendamento e Atendimento, facilitando identificação e contexto para tutores e equipe.

Por fim, é possível configurar Turnos disponíveis (Manhã, Tarde, Noite).

Essas marcações alimentam diretamente a Agenda Inteligente, que calcula capacidade e disponibilidade com base nos períodos configurados.

Edição de colaborador

Na edição, todos os dados podem ser ajustados, incluindo função, bio, especialidades e turnos. É útil para mudanças de atuação, atualizações de cargo ou evolução na carreira clínica. A estrutura permanece igual à criação, permitindo ajustes rápidos e seguros – sem recriar usuário.

Editar colaborador

Ajuste nome e papel de acesso conforme a rotina clínica.

Nome completo	E-mail institucional
Rodrigo Faria Montenegro	rodrigo.montenegro@auravet.com.br
Função	
Contador	
Perfil clínico	Preencha os dados apresentados nos formulários de agendamento e atendimento.
Especialidade clínica	CRMV
Controladoria e Planejamento Financeiro	Registro profissional
Opcional	Opcional
Bio clínica	Cuida da saúde financeira da Auravet, projeções, fluxo de caixa e relacionamento com o escritório fiscal.
Opcional	
Turnos disponíveis	Selecione os períodos em que o colaborador pode ser escalado.
<input checked="" type="checkbox"/> Manhã <input checked="" type="checkbox"/> Tarde <input type="checkbox"/> Noite <small>Deixe todos desmarcados para uma agenda flexível.</small>	

Cancelar **Salvar alterações**

Figura 120. Edição de colaborador

Desativação do colaborador

O botão Desativar remove o acesso do colaborador imediatamente, preservando: histórico de atendimentos, agendas passadas, serviços realizados, vínculos com pets.

Dessa forma, o sistema mantém rastreabilidade completa, mesmo quando alguém deixa a equipe.

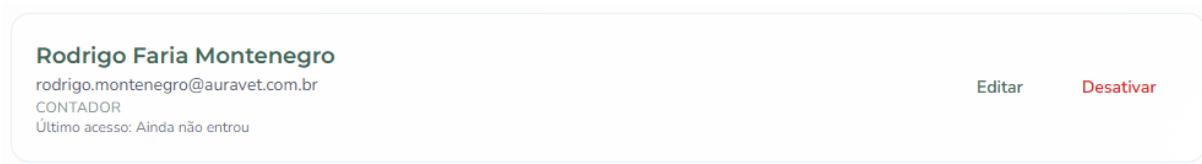


Figura 121. Card do colaborador

2.11.14. Funções

O módulo de Funções é o coração da governança de acesso da Auravet. É aqui que a clínica organiza cada papel interno da equipe, definindo exatamente quem pode ver, editar ou gerenciar cada parte do sistema. A lógica é simples: a função determina o nível de acesso, e os usuários herdam automaticamente esses módulos quando são associados a ela. Isso torna a gestão muito mais segura, previsível e fácil de manter conforme a equipe cresce.

Funções cadastradas

Logo ao entrar na tela, o usuário encontra uma visão clara de todas as funções já criadas na plataforma – como Contador, Enfermeiro, Assistente Administrativo e tantas outras que a clínica pode configurar livremente.

Funções e módulos

Estruture o acesso da equipe definindo quais módulos cada função pode acessar na plataforma Auravet.

Nova função

Funções cadastradas

Organize a equipe conforme responsabilidades e mantenha os módulos sob controle.

Função	Descrição	Ajustar módulos	Desativar	Remover
Contador CONTADOR	Função sem descrição cadastrada. Função ativa • 3 módulos ativos	Ajustar módulos	Desativar	Remover
Assistente Administrativo ASSISTENTE_ADMINISTRATIVO	Função sem descrição cadastrada. Função ativa • 6 módulos ativos	Ajustar módulos	Desativar	Remover
Enfermeiro ENFERMEIRO	Função sem descrição cadastrada. Função ativa • 7 módulos ativos	Ajustar módulos	Desativar	Remover

Contador

Função sem descrição cadastrada.
Função ativa • 3 módulos ativos

Assistente Administrativo

Função sem descrição cadastrada.
Função ativa • 6 módulos ativos

Enfermeiro

Função sem descrição cadastrada.
Função ativa • 7 módulos ativos

Figura 122. Lista de funções e módulos

Cada função aparece acompanhada de três informações fundamentais: o nome, o identificador interno (slug) e uma breve descrição opcional, usada para orientar administradores sobre o uso daquela função.

Além disso, cada função exibe quantos módulos ela possui ativos e mostra visualmente esses módulos em pequenas etiquetas, permitindo uma leitura rápida do que aquele papel consegue acessar no ecossistema Auravet. Essa visualização facilita muito o entendimento e a auditoria das responsabilidades internas, evitando permissões excessivas ou acessos indevidos.

Cada cartão de função oferece duas ações importantes: Ajustar módulos, que abre o painel de permissões editáveis, e os botões Desativar ou Remover, garantindo total controle sobre o ciclo de vida das funções.

Nova função

Ao clicar em **Nova função**, o administrador acessa um formulário completo para estruturar um novo papel interno.

Primeiro, é informado o nome amigável da função, seguido do *slug*, que funciona como seu identificador técnico único. Esse slug é útil para integrações, automações e padronização interna.

Cadastrar nova função

Defina um nome amigável, um identificador único e quais módulos estarão disponíveis para esta função.

Nome da função
Ex.: Coordenador Clínico

Identificador (slug)
EXEMPLO_FUNCAO
Use letras maiúsculas, números e _.

Descrição
Descreva quando utilizar esta função

Opcional

Módulos iniciais (0)

- Administrar usuários**
Permite criar e gerenciar usuários e funções.
- Caixa**
Permite gerenciar contas a receber e registrar pagamentos.
- Gerenciar animais**
Permite cadastrar e editar animais.
- Gerenciar produtos**
Permite cadastrar produtos e ajustar estoque.
- Gerenciar serviços**
Permite registrar e atualizar serviços.

Criar função

Figura 123. Tela de cadastro de função

Uma descrição opcional ajuda a documentar quando essa função deve ser utilizada e qual é seu propósito na rotina da clínica.

Logo abaixo, o sistema apresenta todos os módulos disponíveis. Cada módulo representa uma fronteira de acesso clara – como Gerenciar animais, Visualizar serviços, Administrar usuários, Caixa, Gerenciar produtos, entre outros. O administrador apenas seleciona o que faz sentido para a função, montando o pacote ideal de permissões.

Ao salvar, a nova função já aparece automaticamente na lista principal, pronta para ser associada a colaboradores.

Ajustar módulos de uma função existente

Quando a necessidade muda, ajustar as permissões também é simples. Ao clicar em Ajustar módulos, abre-se uma listagem completa dos módulos disponíveis, cada um acompanhado de uma breve descrição sobre o que libera dentro do sistema.

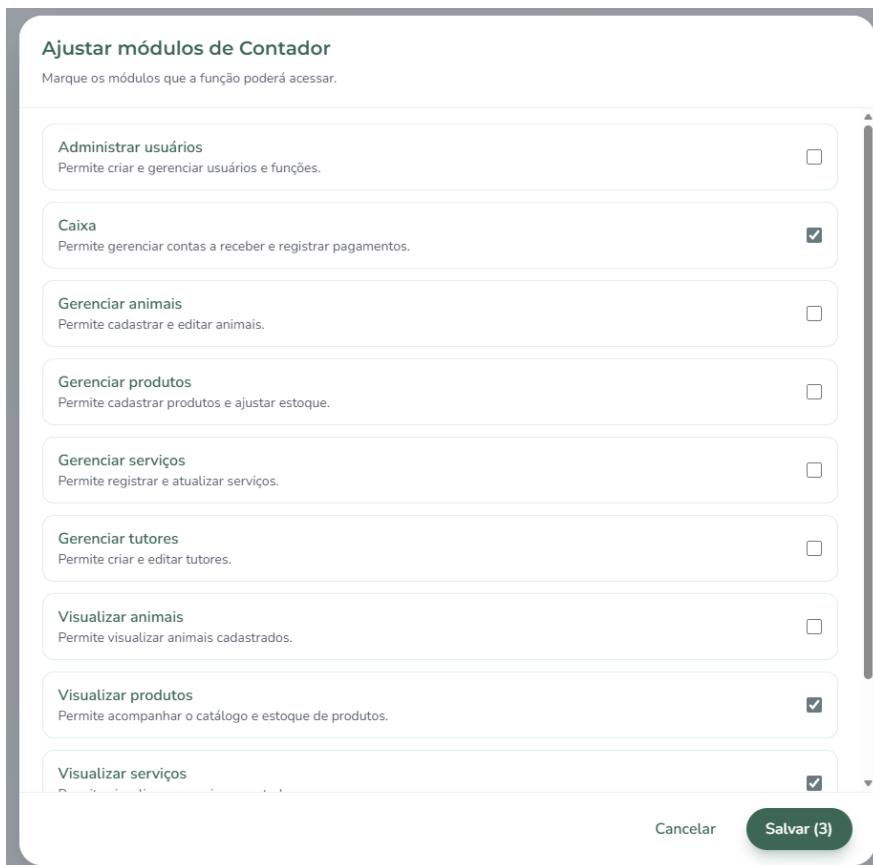


Figura 124. Ajuste de módulos

O administrador pode adicionar novos acessos, retirar permissões ou reorganizar tudo conforme o papel evolui. Nada precisa ser recriado: basta marcar e desmarcar os módulos desejados. Ao salvar, todos os colaboradores associados àquela função passam a herdar as novas permissões automaticamente.

Essa flexibilidade permite que a Auravet acompanhe o crescimento da equipe, ajustes internos e novas responsabilidades sem confusão ou retrabalho.

Visão geral da função

Ao retornar à lista principal, cada função atualizada mostra exatamente quantos módulos estão ativos e quais são eles. As etiquetas coloridas dão um panorama imediato do escopo de acesso daquele papel, reforçando a transparência e o controle das permissões internas.



Figura 125. Card da função

Assim, a clínica mantém uma estrutura segura, organizada e aderente às melhores práticas de gestão de acesso – garantindo que cada colaborador veja apenas o que realmente precisa para trabalhar.

3. Conclusão

A construção do ecossistema Auravet consolida uma visão moderna, integrada e profundamente orientada à experiência – tanto da equipe clínica quanto dos tutores e dos próprios pets. A documentação apresentada demonstra como cada módulo do sistema se conecta para formar um fluxo operacional contínuo, transparente e inteligente, desde o cadastro inicial até o atendimento final, passando pela gestão de agenda, capacidade, usuários, funções e todos os recursos que sustentam o dia a dia da clínica.

Ao detalhar casos de uso, fluxos essenciais e diagramas de sequência, foi possível evidenciar o desenho de um sistema coerente, escalável e seguro, que respeita as necessidades reais do ambiente veterinário. Cada módulo cumpre um papel específico, mas todos trabalham juntos para entregar precisão nas informações, agilidade nas operações e confiabilidade nos registros clínicos. Esse alinhamento evita retrabalho, reduz erros humanos, melhora a comunicação interna e ainda fortalece a experiência do tutor, que passa a perceber a Auravet como um ambiente organizado, profissional e acolhedor.

A documentação também estabelece as bases para evoluções futuras. Com o domínio claro das responsabilidades de cada componente, a expansão do sistema – seja com novos indicadores, fluxos automatizados, prontuários completos ou

integrações externas – passa a ser um processo natural e sustentável. O sistema não apenas resolve problemas presentes; ele foi desenhado para crescer com a clínica.

Em resumo, a Auravet se apoia em um conjunto sólido de processos digitais, construídos com foco em organização, eficiência e cuidado. A documentação aqui concluída serve como mapa técnico e conceitual, guiando todos os envolvidos – desenvolvedores, gestores e equipe clínica – na manutenção e evolução desse ambiente tão essencial. A partir dela, todos falam a mesma língua, compreendem o mesmo fluxo e atuam com segurança sobre a mesma base.

