

EDSON PEREIRA DE CARVALHO

FRANCIANE RAMOS FRANCO

RODRIGO YOSHIDA LOMBEZZI

LUMENSYS

Sistema de Gerenciamento de Planos Funerários

EDSON PEREIRA DE CARVALHO

FRANCIANE RAMOS FRANCO

RODRIGO YOSHIDA LOMBEZZI

LUMENSYS

Sistema de Gerenciamento de Planos Funerários

Projeto Integrador apresentado à Faculdade de Tecnologia Professor José Camargo - Fatec Jales, como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Jefferson Antônio Ribeiro Passarini

Jales

2025

EDSON PEREIRA DE CARVALHO

FRANCIANE RAMOS FRANCO

RODRIGO YOSHIDA LOMBEZZI

LUMENSYS

Sistema de Gerenciamento de Planos Funerários

Projeto Integrador apresentado à Faculdade de Tecnologia Professor José Camargo - Fatec Jales, como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Banca Examinadora:

Prof. (Orientador)

Fatec Jales

Prof.

Fatec Jales

Prof.

Fatec Jales

Jales, ____ de _____ de 2025.

RESUMO

Palavras-chave:

ABSTRACT

Keywords:

Lista de Figuras

| | |
|--|-----|
| Figura 1 – Painel de gestão da TOTVS | 13 |
| Figura 2 – Cadastro de Clientes da Lexos | 14 |
| Figura 3 – Painel de Gestão de Planos Funerários São Paulo | 15 |
| Figura 4 – Diagrama de Classes | 21 |
| Figura 5 – Diagrama de Atores | 31 |
| Figura 6 – Diagrama de Caso de Uso Geral – Visão Administrador | 45 |
| Figura 7 – Diagrama de Caso de Uso Geral – Visão Gerente | 46 |
| Figura 8 – Diagrama de Caso de Uso Geral – Visão Funcionário | 47 |
| Figura 9 – Diagrama de Caso de Uso Individual – Administrador Lista Tanatopraxia | 48 |
| Figura 10 – Diagrama de Caso de Uso Individual – Administrador Lista Cremações | 53 |
| Figura 11 – Diagrama de Caso de Uso Individual – Administrador Lista Transporte | 57 |
| Figura 12 – Diagrama de Sequência – Administrador Cadastra Tanatopraxia | 61 |
| Figura 13 – Diagrama de Sequência – Administrador Altera Tanatopraxia | 62 |
| Figura 14 – Diagrama de Sequência – Administrador Lista Tanatopraxia | 63 |
| Figura 15 – Diagrama de Máquina de Estado – Gerenciamento de Contrato | 64 |
| Figura 16 – Persona 1 | 68 |
| Figura 17 – Persona 2 | 69 |
| Figura 18 – Wireframe – Tela Home | 71 |
| Figura 19 – Wireframe – Tela de Login | 72 |
| Figura 20 – Wireframe – Tela de Cadastro de Usuário | 73 |
| Figura 21 – Wireframe – Tela de Cadastro de Empresa | 74 |
| Figura 22 – Wireframe – Tela Dashboard | 75 |
| Figura 23 – Wireframe – Tela de Gerenciamento de Planos | 76 |
| Figura 24 – Protótipo – Tela home | 77 |
| Figura 25 – Protótipo – Tela de login | 77 |
| Figura 26 – Protótipo – Tela de Cadastro de Usuário | 79 |
| Figura 27 – Protótipo – Tela de Cadastro de Empresa | 79 |
| Figura 28 – Protótipo – Tela Dashboard | 81 |
| Figura 29 – Protótipo – Tela de Gerenciamento de Planos | 82 |
| Figura 30 – Modelo Entidade Relacionamento | 84 |
| Figura 31 – Diagrama de Implantação | 100 |

Lista de Quadros

| | |
|---|----|
| Quadro 1 – Requisitos Funcionais | 16 |
| Quadro 2 – Requisitos Não Funcionais | 18 |
| Quadro 3 – Descrição Classe Empresa | 22 |
| Quadro 4 – Descrição Classe Cliente | 23 |
| Quadro 5 – Descrição Classe PlanosFunerarios | 23 |
| Quadro 6 – Descrição Classe Pessoa Falecida | 24 |
| Quadro 7 – Descrição Classe Descrição do Enum EstadoCivil | 24 |
| Quadro 8 – Descrição do Enum Sexo | 25 |
| Quadro 9 – Descrição Classe Transporte | 25 |
| Quadro 10 – Descrição Classe Cremação | 25 |
| Quadro 11 – Descrição Classe Velório | 26 |
| Quadro 12 – Descrição Classe User | 26 |
| Quadro 13 – Descrição Enum Tipo Funcionário | 27 |
| Quadro 14 – Descrição Classe Contrato | 27 |
| Quadro 15 – Descrição Classe Benefícios | 28 |
| Quadro 16 – Descrição Classe Parcelamento | 28 |
| Quadro 17 – Descrição Classe Tanatopraxia | 29 |
| Quadro 18 – Descrição Enum RelacaoDependente | 29 |
| Quadro 19 – Descrição Enum FormaPagamento | 30 |
| Quadro 20 – Descrição Classe BenefíciosContrato | 30 |
| Quadro 21 – Lista de mensagens | 32 |
| Quadro 22 – Lista de casos de uso | 32 |
| Quadro 23 – Documentação – Listar Tanatopraxia | 49 |
| Quadro 24 – Documentação – Cadastrar Tanatopraxia | 49 |
| Quadro 25 – Documentação – Alterar Tanatopraxia | 51 |
| Quadro 26 – Documentação – Listar Cremação | 53 |
| Quadro 27 – Documentação – Cadastrar Cremação | 54 |
| Quadro 28 – Documentação – Alterar Cremação | 55 |
| Quadro 29 – Documentação – Listar Transporte | 57 |
| Quadro 30 – Documentação – Cadastrar Transporte | 58 |
| Quadro 31 – Documentação – Alterar Transporte | 59 |
| Quadro 32 – Cenário – Registro de Novo Plano Funerário | 66 |

| | |
|---|----|
| Quadro 33 – Cenário – Consulta e Validação de Plano Funerário Pós-Óbito | 66 |
| Quadro 34 – Tabela Empresa | 85 |
| Quadro 35 – Tabela Plano Funerário | 86 |
| Quadro 36 – Tabela Usuário | 86 |
| Quadro 37 – Tabela Cliente | 87 |
| Quadro 38 – Tabela Contrato | 88 |
| Quadro 39 – Tabela Parcelamento | 88 |
| Quadro 40 – Tabela Pessoa Falecida | 89 |
| Quadro 41 – Tabela Dependente | 89 |
| Quadro 42 – Tabela Cremação | 89 |
| Quadro 43 – Tabela Tanatopraxia | 90 |
| Quadro 44 – Tabela Transporte | 90 |
| Quadro 45 – Tabela Velório | 91 |
| Quadro 46 – Tabela Benefício | 91 |
| Quadro 47 – Tabela BeneficioContrato | 91 |
| Quadro 48 – Softwares Utilizados | 94 |

SUMÁRIO

| | | |
|-------|--|----|
| 1 | INTRODUÇÃO | 10 |
| 2 | LEVANTAMENETO DE REQUISITOS | 10 |
| 2.1 | DESCRIÇÃO DE OBJETIVOS DOS SISTEMA | 11 |
| 2.2 | ANÁLISE DE SISTEMAS EXISTENTES | 11 |
| 2.3 | DESCRIÇÃO DOS PRINCIPAIS PROBLEMAS | 15 |
| 2.4 | DESCRIÇÃO DOS REQUISITOS FUNCIONAIS | 15 |
| 2.5 | DESCRIÇÃO DOS REQUISITOS NÃO FUNCIONAIS..... | 17 |
| 3 | VISÃO DE CASO DE USO – UML | 18 |
| 3.1 | DIAGRAMA DE CLASSES | 20 |
| 3.2 | DICIONÁRIO DE CLASSES | 20 |
| 3.3 | DEFINIÇÃO DOS ATORES | 30 |
| 3.4 | LISTA CASO DE USO | 30 |
| 3.5 | DIAGRAMA DE CASO DE USO | 42 |
| 3.6 | DIAGRAMA DE CASO DE USO INDIVIDUAIS | 45 |
| 3.6.1 | DIAGRAMA DE CASO DE USO INDIVIDUAL – ADMINISTRADOR LISTA TANATOPRAXIA | 46 |
| 3.6.2 | DIAGRAMA DE CASO DE USO INDIVIDUAL – ADMINISTRADOR LISTA CREMAÇÕES..... | 50 |
| 3.6.3 | DIAGRAMA DE CASO DE USO INDIVIDUAL – ADMINISTRADOR LISTA TRANSPORTE..... | 54 |
| 3.7 | DIAGRAMA DE CASO DE SEQUÊNCIA | 57 |
| 3.8 | DIAGRAMA DE MÁQUINA DE ESTADO..... | 61 |
| 4 | DEFINIÇÃO DE INTERFACE DO USUÁRIO (UX)..... | 62 |
| 4.1 | DESCRIÇÃO DE CENÁRIO..... | 62 |
| 4.2 | DESCRIÇÃO DE PERSONAS | 64 |
| 4.3 | ESBOÇOS DE TELAS (WIREFRAMES)..... | 66 |
| 4.4 | PROTÓTIPO DE TELA | 73 |
| 5 | BANCO DE DADOS | 78 |
| 5.1 | MODELOS DE ENTIDADE RELACIONAMENTO | 78 |
| 5.2 | SCRIPT DE TABELAS..... | 80 |
| 5.3 | MAPEAMENTO DE OBJETO RELACIONAL – ORM | 88 |
| 6 | ARQUITETURA DE SOFTWARE | 88 |
| 6.1 | ARQUITETURA DE DESENVOLVIMENTO | 89 |

| | |
|-----------------------------------|-----|
| 6.1.1 BACK-END..... | 92 |
| 6.1.2 FRONT-END..... | 92 |
| 6.2 SEGURANÇA DA INFORMAÇÃO | 94 |
| 6.3 IMPLANTAÇÃO DO SISTEMA | 95 |
| 7 CONCLUSÃO | 98 |
| 8 REFERÊNCIAS..... | 100 |

1 INTRODUÇÃO

A organização de serviços funerários e a gestão de planos funerários têm se tornado cada vez mais essenciais para garantir previsibilidade, segurança e tranquilidade aos clientes e seus familiares em momentos delicados. Empresas que atuam nesse setor precisam administrar uma série de processos, como a gestão de contratos, controle de pagamentos, atendimento aos beneficiários, organização de serviços prestados e cumprimento de normas regulatórias. Diante dessa complexidade, a tecnologia surge como uma aliada indispensável na otimização desses processos, possibilitando maior eficiência, redução de erros e aprimoramento do relacionamento com os clientes.

Nos últimos anos, a digitalização dos serviços funerários tem se mostrado uma tendência crescente, impulsionada pela necessidade de automação e pela busca por soluções que tornem os processos mais ágeis e acessíveis. Assim, os sistemas informatizados de gerenciamento de planos funerários desempenham um papel fundamental na modernização das operações, permitindo que empresas do ramo ofereçam um atendimento mais transparente, organizado e personalizado aos seus clientes.

O presente trabalho tem como objetivo desenvolver um sistema de gerenciamento de planos funerários que permita uma administração integrada e eficiente dos serviços. A proposta visa proporcionar às empresas do setor uma ferramenta capaz de otimizar tarefas operacionais, facilitar o acesso a informações relevantes e oferecer uma interface intuitiva que melhore a experiência do usuário. Além disso, busca-se garantir conformidade com as exigências legais e regulatórias, assegurando que os processos sejam conduzidos de forma ética e segura.

Ao longo deste estudo, serão abordadas as principais características de um sistema de gestão eficiente, analisando tecnologias disponíveis, requisitos necessários e benefícios esperados. A pesquisa também irá explorar os desafios enfrentados pelas empresas do setor e as soluções inovadoras que podem contribuir para um gerenciamento mais eficaz e acessível, beneficiando tanto as organizações quanto os clientes atendidos.

2 LEVANTAMENTO DE REQUISITOS

2.1 DESCRIÇÃO DE OBJETIVOS DOS SISTEMA

O sistema de gerenciamento de funerária tem como objetivo principal automatizar os processos operacionais da empresa, aumentando a eficiência e reduzindo a necessidade de intervenção manual. Ele permite o registro e acompanhamento das informações dos clientes, bem como o gerenciamento completo dos planos funerários contratados, com base em contratos personalizados que definem os serviços e benefícios incluídos para cada cliente.

O sistema também organiza todas as etapas relacionadas aos serviços funerários, incluindo o planejamento de funerais, a realização de cremações e o transporte do corpo. Essas funcionalidades garantem que os procedimentos sejam executados conforme o planejado, respeitando prazos, normas e as preferências dos familiares, com maior agilidade e precisão, assim ajudando na padronização e na qualidade da prestação dos serviços.

O sistema garante que todas as operações estejam em conformidade com as leis e regulamentações locais e nacionais, mantendo registros adequados e assegurando o cumprimento dos padrões de qualidade e segurança.

2.2 ANÁLISE DE SISTEMAS EXISTENTES

A gestão eficiente de serviços funerários é fundamental para garantir a organização e a qualidade na administração de planos funerários, cerimônias e processos financeiros. Um sistema bem estruturado possibilita o controle de informações, automação de tarefas e a otimização dos recursos disponíveis, assegurando maior transparência e eficiência operacional.

O sistema Gestão Deathcare by Duofy da TOTVS é uma solução desenvolvida para a gestão de operações funerárias. Ele oferece funcionalidades que incluem a integração com o ERP TOTVS, permitindo uma gestão centralizada e a automação de processos, eliminando a necessidade de controles em papel e múltiplos sistemas de gestão. O sistema pode ser adaptado às necessidades específicas de cada operação, é escalável e pode operar tanto em nuvem quanto na rede de servidores da empresa. Além disso, inclui ferramentas de comunicação como portais e aplicativos personalizáveis com a marca da empresa e ferramentas para gestão de cobranças e negociações. A plataforma também permite a realização de vendas digitais e o uso de tecnologia móvel para cadastro de clientes e fechamento de negócios.

Figura 1 – Painel de gestão da TOTVS

Uso do Plano para Humano - 000002 - Plano Top

Dados do Contrato

Dados Titular Endereço Endereço Cobrança Financeiro Auditoria Seguro Responsável Financeiro

Dados do Plano

Plano* 000002 Descrição* PLANO TOP

Observação

- > Cremação ou sepultamento
- > Serviço funerário
- > Cerimonial
- > Transportado de até 100km

Beneficiários do Contrato Produtos e Serviços Aportamento de Serviços Financeiro Mensagens Histórico Transf. Titularidade Comissões do Vendedor

Beneficiários

| Item | Grau parente | Tipo | Dt Inclusão | Carencia | Nome | CPF | Cobranças Adicionais |
|------|--------------|--------------|-------------|------------|------|------------|----------------------|
| 001 | OUTROS | Titular | 08/10/2024 | 06/04/2025 | | 08/03/2025 | |
| 002 | FILHO | Beneficiário | 08/10/2024 | 06/04/2025 | | 11/05/2025 | |

Resumo Financeiro

| Valor Atual | Última Negociação |
|-------------|-------------------|
| R\$ 78,45 | R\$ 78,45 |
| R\$ 21,44 | R\$ 21,44 |
| R\$ 0,00 | R\$ 0,00 |
| R\$ 99,89 | R\$ 99,89 |

Situação Financeira

ADIMPLENTE

Fonte: TOTVS, 2025

O sistema para funerárias da Lexos ERP é uma solução projetada para gerenciar e controlar de forma eficiente as operações funerárias. Ele oferece funcionalidades como controle detalhado de clientes, incluindo gestão de crédito, grupos de clientes, promoções, cadastro de dependentes e consultas inteligentes. O sistema também permite a emissão de carnês e cobrança bancária, com baixa automática dos pagamentos e comunicação com o banco via arquivos de remessa e retorno.

Além disso, a Lexos ERP oferece gestão de contas a pagar e receber, simplificando o registro de contas e a geração de contas a partir de compras registradas. A plataforma também inclui a emissão de relatórios gerenciais detalhados, proporcionando uma visão abrangente das operações e facilitando a tomada de decisões.

Figura 2 – Cadastro de Clientes da Lexos

The screenshot displays the 'Cadastro de clientes' (Client Registration) window within the Lexos ERP system. The window is titled 'Cadastro de clientes' and includes a 'Histórico' (History) tab. The form contains the following fields and options:

- Código:** 00829
- Física / Jurídica:** Física (dropdown menu)
- Nome:** Bianca Carvalho
- CPF:** (empty field)
- RG:** (empty field)
- Data de Nascimento:** (empty field)
- Sexo:** (empty dropdown menu)
- Categoria:** REVENDA (dropdown menu)
- Estado Civil:** (empty dropdown menu)
- Data da 1ª Compra:** (empty field)

A button labeled 'Análise de Crédito' (Credit Analysis) is located to the right of the form. The background shows the Lexos ERP interface with a sidebar menu and a top navigation bar.

Fonte: Lexos ERP, 2025

O Plano Funerário São Paulo oferece uma gama de planos adaptáveis, com opções para diferentes necessidades e orçamentos. A gestão dos contratos é simplificada, e o sistema proporciona cobertura para titulares e dependentes, incluindo serviços como cremação e atendimento 24 horas. Esse modelo é valioso para a construção de um sistema que prioriza a eficiência e a clareza na comunicação com o cliente.

Com planos flexíveis e acesso facilitado, o Plano Funerário São Paulo garante cobertura completa para seus clientes, incluindo serviços de cremação e suporte contínuo. O sistema de gestão de contratos e pagamentos é intuitivo e descomplicado, tornando-se uma referência na criação de soluções que promovem transparência e facilidade na adesão.

Figura 3 – Gestão de Planos Funerários São Paulo

| <div> <div>MELHOR VALOR</div> <div>PLANO FUNERÁRIO LUNA BRONZE</div> </div> | PLANO FUNERÁRIO LUNA PRATA | PLANO FUNERÁRIO LUNA OURO | PLANO FUNERÁRIO LUNA TOP |
|---|--|--|--|
| <div>R\$ 69,00/mês</div> | <div>R\$ 89,00/mês</div> | <div>R\$ 129,00/mês</div> | <div>CONSULTE VALORES PROMOCIONAIS</div> |
| <div>Carência 180 dias</div> | <div>Carência 180 dias</div> | <div>Carência 180 dias</div> | <div>Plano sem carência</div> |
| <div>Titular até 65 anos incompletos</div> | <div>Titular até 65 anos incompletos</div> | <div>Titular até 70 anos incompletos</div> | <div>Titular sem limite de idade</div> |
| <div>Cônjuge até 65 anos incompletos</div> | <div>Cônjuge até 65 anos incompletos</div> | <div>Cônjuge até 70 anos incompletos</div> | <div>Cônjuge sem limite de idade</div> |
| <div>Filhos até 24 anos</div> | <div>4 Familiares até 70 anos</div> | <div>8 Familiares sem limite de idade, do mesmo grau de parentesco</div> | <div>Filhos sem limite de idade</div> |
| <div>1 familiar até 65 anos</div> | <div>2 Idosos até 80 anos</div> | <div>Uma Mortuária (caixão)</div> | <div>Pai e mãe sem limite de idade</div> |
| <div>Uma Mortuária (caixão)</div> | <div>Uma Mortuária (caixão)</div> | <div>Cremação</div> | <div>Sogra e sogra sem limite de idade</div> |
| <div>(1) Coroa com flores do campo ou flores da época</div> | <div>Cremação</div> | <div>(1) Coroa com flores do campo ou flores da época</div> | <div>Uma Mortuária (caixão)</div> |
| <div>Taxa de velório e taxa de sepultamento em cemitérios municipais</div> | <div>(1) Coroa com flores do campo ou flores da época</div> | <div>Taxa de velório e taxa de sepultamento em cemitérios municipais</div> | <div>Cremação</div> |
| <div>Agente Social para assistência a família</div> | <div>Taxa de velório e taxa de sepultamento em cemitérios municipais</div> | <div>Agente Social para assistência a família</div> | <div>(1) Coroa com flores do campo ou flores da época</div> |
| <div>Enfeite Floral. Véu rendado. Paramentações</div> | <div>Agente Social para assistência a família</div> | <div>Enfeite Floral. Véu rendado. Paramentações</div> | <div>Taxa de velório e taxa de sepultamento em cemitérios municipais</div> |
| <div>Remoção Entre Ida e volta - Transporte até 300 km</div> | <div>Enfeite Floral. Véu rendado. Paramentações</div> | <div>Remoção Entre Ida e volta - Transporte até 300 km</div> | <div>Agente Social para assistência a família</div> |
| <div>Cobertura Nacional</div> | <div>Remoção Entre Ida e volta - Transporte até 300 km</div> | <div>Cobertura Nacional</div> | <div>Enfeite Floral. Véu rendado. Paramentações</div> |
| <div>Plano Pet com Cremação - Promoção por tempo indeterminado</div> | <div>Cobertura Nacional</div> | <div>Plano Pet com Cremação - Promoção por tempo indeterminado</div> | <div>Remoção Entre Ida e volta - Transporte até 300 km</div> |
| | <div>Plano Pet com Cremação - Promoção por tempo indeterminado</div> | | <div>Cobertura Nacional</div> |

Fonte: Plano funerário São Paulo, 2025

2.3 DESCRIÇÃO DOS PRINCIPAIS PROBLEMAS

Os sistemas de gerenciamento de funerárias enfrentam diversos desafios específicos. Entre os principais problemas estão as falhas de comunicação, que resultam em erros operacionais e retrabalho, e a falta de integração entre sistemas, o que gera informações divergentes e dificulta a gestão centralizada.

Outro principal problema é na dificuldade em atrair e gerenciar potenciais clientes, o que pode levar à perda de oportunidades de venda. A ausência de um sistema automatizado também contribui para uma qualificação inconsistente dessas oportunidades, provocando desorganização no funil de vendas e dificultando o acompanhamento eficaz. Além disso, a falta de organização no gerenciamento dos planos funerários contratados por cada cliente pode gerar confusões sobre os serviços contratados, prazos e benefícios previstos em contrato.

A falta de um sistema automatizado também contribui para uma qualificação inconsistentes de oportunidades, causando desorganização no funil de vendas. Por fim, a ausência de um sistema integrado impede análises precisas e a identificação de áreas de melhoria.

2.4 DESCRIÇÃO DOS REQUISITOS FUNCIONAIS

Sommerville (2018), explica que requisitos funcionais são as especificações gerais sobre o que um sistema deve realizar. Esses requisitos “devem descrever em detalhes as funções do sistema, suas entradas, saídas e exceções” (Sommerville, 2018, p.89).

Os requisitos podem ser divididos em duas categorias principais. Requisitos para usuários, que são direcionados aos usuários finais e gerentes do sistema, descrevendo as funcionalidades do sistema sob a ótica do usuário. Requisitos do sistema, destinados aos desenvolvedores, detalham as funcionalidades do sistema do ponto de vista técnico (Sommerville, 2018).

Embora os requisitos funcionais geralmente descrevam o que o sistema deve fazer, “em alguns casos, [...] também podem declarar explicitamente o que o sistema não deve fazer” (Sommerville, 2018, p.89). No contexto do Sistema de Gerenciamento de Funerárias, pode-se verificar os seguintes requisitos, conforme o Quadro 1.

Quadro 1 – Requisitos Funcionais

| | Requisitos Funcionais | Descrição |
|--|-----------------------|-----------|
|--|-----------------------|-----------|

| | | |
|----|---------------------------------|--|
| 1 | Cadastro de Clientes | O sistema deve permitir o registro de informações dos clientes, incluindo dados pessoais, contatos, preferências, histórico de serviços e plano funerário escolhido. |
| 2 | Alterar Clientes | O sistema deve permitir que o gerente altere informações dos clientes. |
| 3 | Listagem de Cliente | O sistema deve permitir lista todos os clientes cadastrados no sistema. |
| 4 | Controle de acesso | O sistema deve verificar as credenciais do usuário. |
| 5 | Cadastro de Empresas funerárias | O sistema deve permitir que o administrador cadastre as empresas funerárias. |
| 6 | Cadastro de Funcionários | O sistema deve permitir que o gerente cadastre os funcionários que trabalham na funerária e permite que defina o cargo do funcionário. |
| 7 | Registro de Dados do Falecido | O sistema deve registrar os dados do falecido. |
| 8 | Cadastro Velório | O sistema deve permitir o cadastro um velório de um falecido. |
| 9 | Cadastro de Cremação | O sistema deve registrar a cremação de uma pessoa falecida. |
| 10 | Cadastro de Transporte | O sistema deve registrar informações sobre o transporte do corpo. |
| 11 | Cadastro Planos Funerários | O sistema deve registrar criar planos funerários. |
| 12 | Alteração Planos Funerários | O sistema deve permitir que o gerente faça alteração dos dados dos planos funerários. |
| 13 | Cadastro de Tanatopraxia | O sistema deve permitir que o registro e controle dos procedimentos de tanatopraxia realizados pela funerária. |
| 14 | Gerenciamento de Parcelamentos | O sistema deve permitir o controle e a administração dos parcelamentos dos pagamentos relacionados aos contratos dos planos funerários. |
| 15 | Cadastro de Contratos | O sistema deve permitir o registro dos contratos entre a funerária e os clientes, detalhando os serviços e condições acordadas. |
| 16 | Alterar Contrato | O sistema deve permitir que o gerente atualize ou modifique informações dos contratos cadastrados. |

| | | |
|----|--------------------------------|--|
| 17 | Cadastrar Benefícios | O sistema deve permitir a inclusão dos benefícios vinculados aos planos funerários. |
| 18 | Alterar Benefícios | O sistema deve permitir a edição ou atualização dos benefícios cadastrados no sistema. |
| 19 | Integração com Outros Sistemas | Integrar-se com outros sistemas e plataformas, como bancos de dados de cemitérios e fornecedores de produtos funerários. Facilitar a troca de informações e a colaboração com parceiros externos. |
| 20 | Comunicação com Clientes | Enviar notificações e atualizações aos clientes por meio de e-mails, mensagens de texto ou aplicativos de comunicação. Permitir a comunicação bidirecional, onde os clientes podem fazer perguntas e receber respostas em tempo real. |

Fonte: Elaborado pelos autores.

2.5 DESCRIÇÃO DOS REQUISITOS NÃO FUNCIONAIS

Além dos requisitos funcionais, Sommerville (2018) também aborda os requisitos não funcionais do sistema. Esses requisitos abrangem aspectos gerais do sistema, em vez de se concentrarem em serviços e características individuais como os requisitos funcionais. Alguns exemplos de requisitos não funcionais são “restrições de tempo, restrições sobre o processo de desenvolvimento e restrições impostas por padrões” (Sommerviller, 2018, p. 89).

Os requisitos não funcionais surgem das necessidades dos usuários, que se devem a restrições orçamentárias, políticas organizacionais, necessidade de interoperabilidade com outros sistemas de software ou hardware, ou fatores externos, como normas de segurança (safety) ou legislação relativa à privacidade (Sommerville, 2018, p. 91).

Os requisitos não funcionais de um sistema de gerenciamento de funerárias podem ser variados e devem ser cuidadosamente considerados para assegurar a eficiência, segurança e a adequação do sistema às necessidades específicas desse setor. Alguns exemplos de requisitos não funcionais que podem ser relevantes incluem (Quadro 2).

Quadro 2 – Requisitos Não Funcionais

| Requisitos Não Funcionais | Descrição |
|---------------------------|---|
| | O sistema deve ser capaz de processar transações rapidamente, mesmo durante picos de uso. Isso inclui |

| | |
|------------------|--|
| Desempenho | tempos de resposta rápidos para consultas e atualizações de dados. |
| Segurança | A proteção dos dados dos clientes é crucial. O sistema deve implementar criptografia de dados, autenticação de usuários e controle de acesso para garantir que apenas usuários autorizados possam acessar informações sensíveis. |
| Confiabilidade | O sistema deve ser altamente confiável, com um tempo de inatividade mínimo. Isso pode ser alcançado através de redundância de hardware, backups regulares e um plano de recuperação de desastres. |
| Usabilidade | O sistema deve ser fácil de usar, com uma interface intuitiva que permita aos usuários realizar suas tarefas sem dificuldades. Isso inclui a acessibilidade para usuários com diferentes níveis de habilidade técnica. |
| Manutenibilidade | O sistema deve ser fácil de manter e atualizar. Isso inclui uma arquitetura modular que permita a adição de novas funcionalidades sem afetar o funcionamento existente. |
| Compatibilidade | O sistema deve ser compatível com diferentes dispositivos e plataformas, incluindo desktops, tablets e smartphones. Isso garante que os usuários possam acessar o sistema de qualquer lugar. |
| Regulamentação | O sistema deve estar em conformidade com todas as regulamentações e leis aplicáveis, incluindo aquelas relacionadas à privacidade de dados e proteção ao consumidor. |

Fonte: Elaborado pelos autores.

3 VISÃO DE CASO DE USO – UML

O diagrama de classes é fundamental para o desenvolvimento de um sistema, na UML o diagrama é estruturação lógica do sistema e servindo como base para a construção de outros diagramas.

Seu principal enfoque está em permitir a visualização das classes que compõem o sistema com seus respectivos atributos e métodos, bem como em demonstrar como as classes do diagrama se relacionam, complementam e transmitem informações entre si. Esse diagrama apresenta uma visão estática de como as classes estão organizadas, preocupando-se em como definir a estrutura lógica delas (Guedes, 2018, p.133).

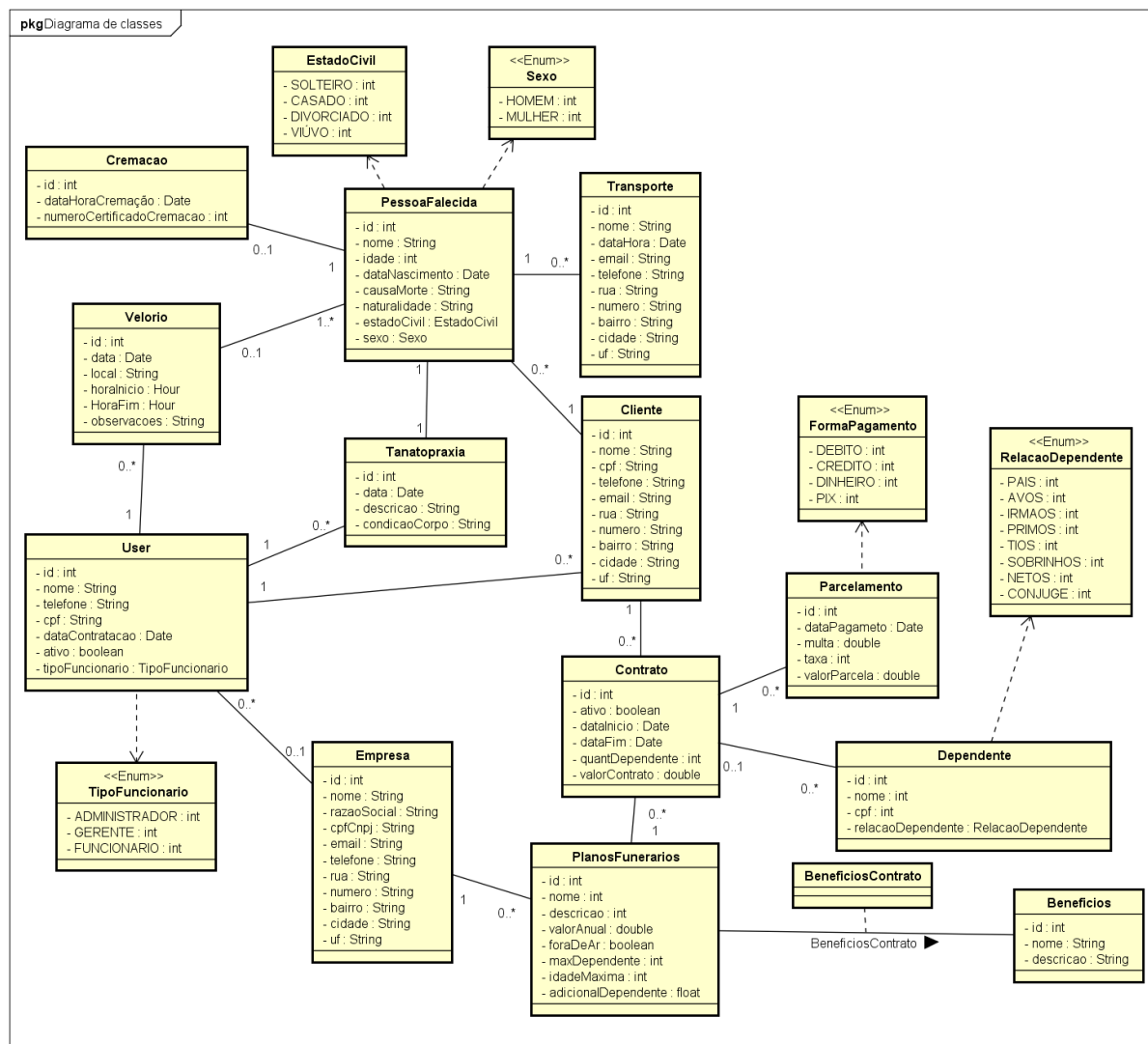
De acordo com Sommerville (2018) os diagramas de classe representam as classes como retângulos, contendo o nome, atributos e seus tipos. As associações entre as classes são indicadas por linhas que conectam esses retângulos.

A classe PlanoFunerario é responsável por representar os planos ofertados pela funerária, incluindo informações como tipo, cobertura, valor e benefícios. Esta classe está diretamente relacionada ao Contrato, uma vez que é por meio deste que o plano é formalmente vinculado a um cliente.

Já a classe PessoaFalecida é utilizada para registrar informações sobre o falecido que está associado a um cliente. Essa classe armazena dados como nome completo, data de falecimento, local do óbito e tipo de serviço aplicado (tanatopraxia, cremação, sepultamento). Ela é essencial para a operação dos serviços, sendo consultada sempre que há necessidade de executar ou revisar procedimentos funerários.

3.1 DIAGRAMA DE CLASSES

Figura 4 – Diagrama de Classes



Fonte: Elaborado pelos autores.

3.2 DICIONÁRIO DE CLASSES

Na UML segundo Guedes (2011), possibilita que os desenvolvedores visualizem os resultados de suas modelagens em diagramas padronizados, o dicionário de classes descreve todas as classes e seus atributos. O diagrama de classes na Figura 4 a classe “PessoaFalecida” é uma das principais classes para gerenciar os serviços funerários, responsável por guardar informações do falecido.

As classes de “Velório”, “Cremacao” e “Tanatopraxia” guardam informações sobre respectivamente os dados do velório como quando ocorreu e a hora, cremação dados como a data e a hora e o número de verificação da cremação e tanatopraxia guarda dados do corpo. A classe de “Usuario” é responsável pelo cadastro e alteração de perfil de usuário, atribuindo permissões e responsabilidades específicas no gerenciamento de serviços funerários, sendo o controle de acesso feito por meio de autenticação, por meio de login e senha.

A classe “Empresa” é necessária para o cadastro de empresas funerárias e gerencia os planos funerários da classe “PlanosFunerarios”, essa classe define os valores a quantidade de dependentes para o plano com base nos benefícios definidos na classe “Benefícios”, onde são definidos a descrição dos benefícios e suas descrições.

Se o plano funerário for acordado com o cliente, a classe “Contrato” terá a função de gerenciar se ainda está ativo e a quantidade de dependentes relacionados no contrato da classe “Dependente”, que faz o cadastro das pessoas que farão parte do plano, para gerenciar os pagamentos de contrato a classe de “Parcelamento” que gerencia o pagamento das parcelas de cada mês. Por fim a classe “Cliente” guarda as informações de cliente que será vinculada ao contrato e pessoa falecida para ter o controle de contrato das pessoas que faleceram que estavam cadastradas como no contrato.

Nesta seção, são apresentados os Dicionários de Classes, que têm como finalidade oferecer uma visão detalhada e clara sobre o funcionamento do sistema de gestão de funerárias. Cada quadro a seguir ilustra uma classe específica, descrevendo seus atributos e suas respectivas funcionalidades no contexto do gerenciamento funeral.

A classe “Empresa” (Quadro 3) é destinada ao registro e gerenciamento das informações das funerárias cadastradas no sistema. Esse registro inclui dados como razão social, CNPJ, endereço e contato, permitindo a organização e a administração centralizada das empresas. Além disso, essa classe desempenha um papel essencial na estruturação do sistema, viabilizando a comunicação eficiente entre as instituições e o acompanhamento detalhado de suas operações no contexto funerário.

Quadro 3 – Descrição Classe Empresa

| Atributo | Tipo | Descrição |
|----------|---------|--|
| id | Integer | Código que irá identificar a empresa cadastrada. |
| cnpj | String | Código que identifica a empresa no cadastro nacional de pessoas jurídicas. |
| nomeEmp | String | Nome de identificação da empresa. |
| emailEmp | String | E-mail de identificação utilizado para acessar o sistema. |

| | | |
|-------------|--------|--------------------------------|
| | | |
| telefoneEmp | String | Número de telefone da empresa. |
| enderecoEmp | String | Endereço da empresa. |

Fonte: Elaborado pelos autores.

O (Quadro 4) apresenta classe “Cliente” é responsável por armazenar as informações dos clientes cadastrados no sistema. Ela permite o gerenciamento dos dados essenciais dos clientes que possuem planos funerários, garantindo a correta identificação e associação aos serviços contratados.

Quadro 4 – Descrição Classe Cliente

| Atributo | Tipo | Descrição |
|-----------|---------|--|
| idCliente | Integer | Código único para identificar o cliente. |
| nome | String | Nome completo do cliente. |
| cpf | String | CPF do cliente, utilizado para identificação. |
| endereco | String | Endereço residencial do cliente. |
| telefone | String | Número de telefone para contato. |
| email | String | E-mail utilizado para contato e acesso ao sistema. |

Fonte: Elaborado pelos autores.

A classe “Planos Funerários” (Quadro 5) é responsável por armazenar e gerenciar as informações dos planos funerários oferecidos pela empresa. Cada plano contém detalhes sobre seus benefícios, valores e cobertura, permitindo que os clientes escolham a melhor opção de acordo com suas necessidades.

Quadro 5 – Descrição Classe Planos Funerários

| Atributo | Tipo | Descrição |
|------------|---------|--|
| idPlano | Integer | Código único para identificar o plano funerário. |
| nomePlano | String | Nome do plano funerário. |
| descricao | String | Descrição dos serviços incluídos no plano. |
| valorPlano | Double | Valor mensal ou total do plano funerário. |

Fonte: Elaborado pelos autores.

A classe “pessoaFalecida” (Quadro 6) é destinada ao registro e gerenciamento das informações referentes aos falecidos no sistema. Essa classe inclui dados como nome completo, data de nascimento, data de falecimento, documentação e outros detalhes relevantes. Sua funcionalidade é essencial para organizar e centralizar os registros, garantindo a precisão das informações e permitindo o acompanhamento eficiente de procedimentos relacionados aos serviços funerários.

Quadro 6 – Descrição Classe Pessoa Falecida

| Atributo | Tipo | Descrição |
|----------------|--------------|--|
| id | Integer | Código único para identificar a pessoa falecida. |
| nome | String | Nome da pessoa falecida. |
| idade | Integer | Idade da pessoa falecida. |
| dataNascimento | Date | Data de nascimento da pessoa falecida. |
| causaMorte | String | Causa da morte da pessoa falecida. |
| naturalidade | String | Onde a pessoa falecida nasceu. |
| estadoCivil | Estado Civil | Estado civil da pessoa falecida. |
| sexo | Sexo | Gênero da pessoa falecida. |

Fonte: Elaborado pelos autores.

A classe “estadoCivil” (Quadro 7) é responsável pelo registro e gerenciamento das informações relacionadas ao estado civil das pessoas cadastradas no sistema. Essa classe abrange detalhes como as categorias de estado civil disponíveis (solteiro, casado, viúvo, divorciado, entre outros), possibilitando a padronização e a organização dos dados. Sua funcionalidade é essencial para garantir a consistência nas informações pessoais, facilitando o uso dessas informações em outros processos do sistema.

Quadro 7 – Descrição do Enum EstadoCivil

| Atributo | Tipo | Descrição |
|------------|---------|--------------------------------------|
| Solteiro | Integer | Identifica se a pessoa é solteira. |
| Casado | Integer | Identifica se a pessoa é casada. |
| Divorciado | Integer | Identifica se a pessoa é divorciada. |
| Viúvo | Integer | Identifica se a pessoa é viúva. |

Fonte: Elaborado pelos autores.

O enum "Sexo" (Quadro 8) define se a pessoa é homem ou mulher

Quadro 8 – Descrição do Enum Sexo

| Atributo | Tipo | Descrição |
|----------|---------|-----------------------------------|
| HOMEM | Integer | Identificar se a pessoa é homem. |
| MULHER | Integer | Identificar se a pessoa é mulher. |

Fonte: Elaborado pelos autores.

A classe "Transporte" (Quadro 9) é responsável pelo registro de quando e onde o corpo do falecido foi transportado, detalhando onde foi a chegada e o horário e o tipo de transporte utilizado para mover o corpo de localização.

Quadro 9 – Descrição Classe Transporte

| Atributo | Tipo | Descrição |
|----------------|----------|--|
| id | Integer | Código único para o transporte. |
| tipoTransporte | Integer | Especifica o tipo de transporte utilizado. |
| dataHora | DateTime | Data e horário em que o transporte será realizado. |
| destino | String | Local de destino do transporte. |

Fonte: Elaborado pelos autores.

A classe "Cremação" (Quadro 10) é projetada para registrar e gerenciar as informações relacionadas aos serviços de cremação oferecidos pelas funerárias cadastradas no sistema. Este registro abrange dados como a data e o horário da cerimônia de cremação, local, nome do falecido e informações de contato dos familiares. Com essa organização, o sistema facilita a administração centralizada das cremações, proporcionando um acompanhamento detalhado dos procedimentos.

Além disso, a classe "Cremação" desempenha um papel essencial na estruturação do sistema, promovendo uma comunicação eficiente entre as funerárias e os familiares, garantindo que todas as necessidades sejam atendidas de forma respeitosa e organizada durante este momento delicado.

Quadro 10 – Descrição Classe Cremação

| Atributo | Tipo | Descrição |
|-------------------|---------|--|
| id | Integer | Código único para identificar a cremação. |
| data | Date | Data programada para a realização da cremação. |
| local | String | Local onde será realizado a cremação. |
| numeroCertificado | String | Identificação do certificado de cremação. |

Fonte: Elaborado pelos autores.

A classe “Velório” (Quadro 11) é destinada ao registro e gerenciamento das informações relacionadas aos velórios realizados pelas funerárias cadastradas no sistema. Esse registro inclui dados como data e horário do velório, local, nome do falecido, e informações de contato dos familiares. Com essa organização, o sistema facilita a administração centralizada dos velórios, proporcionando um acompanhamento detalhado das cerimônias. Além disso, a classe “Velório” desempenha um papel crucial na estruturação do sistema, permitindo uma comunicação eficiente entre as funerárias e os familiares, garantindo que todas as necessidades sejam atendidas de maneira respeitosa e organizada durante este momento delicado.

Quadro 11 – Descrição Classe Velório

| Atributo | Tipo | Descrição |
|----------------|----------|--|
| id | Integer | Código único para identificar o velório. |
| local | String | Data programada para a realização da cremação. |
| dataInicio | DateTime | Data e hora de início do velório. |
| dataFim | DateTime | Data e hora do término do velório. |
| pessoaFalecida | String | Nome da pessoa falecida associada ao velório. |

Fonte: Elaborado pelos autores.

A classe “User” (Quadro 12) é destinada ao registro e gerenciamento das informações dos funcionários das funerárias cadastradas no sistema. Esse registro inclui dados como nome, CPF, cargo, telefone de contato, e endereço. Além disso, podem ser armazenadas informações adicionais como data de admissão, horários de trabalho e certificações profissionais. A organização dessas informações é fundamental para garantir uma administração eficiente e centralizada dos funcionários, facilitando a comunicação entre eles e a empresa, bem como o acompanhamento de suas atividades e desempenho.

Quadro 12 – Descrição Classe User

| Atributo | Tipo | Descrição |
|-----------------|---------|--|
| id | Integer | Código único para identificar o usuário. |
| nome | String | Nome do usuário. |
| cpf | String | Cadastro de pessoa física do usuário. |
| tipoFuncionario | String | Cargo ocupado pelo funcionário da funerária. |
| telefone | String | Informação de contato. |

Fonte: Elaborado pelos autores.

O Enum “TipoFuncionario”(Quadro 13) tem o objetivo de definir o nível de acesso do usuário dentro do sistema, facilitando a gestão de recursos humanos e garantindo que as operações funerárias sejam conduzidas de maneira eficaz e profissional.

Quadro 13 – Descrição Enum Tipo Funcionário

| Atributo | Tipo | Descrição |
|---------------|---------|---------------------------------------|
| Administrador | Integer | Acesso Total do sistema. |
| Gerente | Integer | Acesso Total da parte da funerária. |
| Funcionário | Integer | Acesso Parcial da parte da funerária. |

Fonte: Elaborado pelos autores.

A classe “Contrato” é responsável por registrar e gerenciar os contratos firmados entre os clientes e a funerária. Esses contratos contêm informações como a data de início e fim, status de ativação e o número de dependentes vinculados. Essa estrutura permite o controle dos serviços contratados. A organização contratual também contribui para que o sistema acompanhe a validade dos planos e possibilite renovações, atualizações ou encerramentos conforme necessários.

Quadro 14 – Descrição Classe Contrato

| Atributo | Tipo | Descrição |
|-----------------|----------|---|
| id | Integer | Código único para o contrato. |
| ativo | boolean | Indica se o contrato está ativo ou inativo. |
| dataInicio | DateTime | Data de início do contrato. |
| dataFim | DateTime | Data de término do contrato. |
| quantDependente | Integer | Quantidade de dependentes associados ao contrato. |
| ValorContrato | double | Valor do contrato anual. |

Fonte: Elaborado pelos autores.

A classe “Benefícios” tem a finalidade de cadastrar e gerenciar os serviços ou vantagens oferecidas como parte dos planos funerários. Cada benefício pode incluir auxílios como transporte, descontos, atendimento domiciliar, entre outros. Essa classe organiza os benefícios separadamente, assim permitindo que os contratos e planos possam ser personalizados conforme as necessidades da funerária.

Quadro 15 – Descrição Classe Benefícios

| Atributo | Tipo | Descrição |
|-----------|---------|----------------------------------|
| id | Integer | Código único para o benefício. |
| nome | boolean | Nome do benefício. |
| descrição | String | Detalhes do benefício oferecido. |

Fonte: Elaborado pelos autores.

A classe “Parcelamento” tem o objetivo de controlar os pagamentos dos serviços e contratos funerários. Ela registra datas de vencimento, data de início dos pagamentos, além de multas e taxas aplicáveis. Com essa estrutura, o sistema permite acompanhar o histórico de pagamento dos clientes.

Quadro 16 – Descrição classe Parcelamento

| Atributo | Tipo | Descrição |
|---------------|----------|--|
| id | Integer | Código único do parcelamento. |
| dataPagamento | DateTime | Data programada para pagamento da parcela. |
| valorParcela | double | Valor da parcela. |
| multa | double | Valor da multa aplicada em caso de atraso. |
| taxa | double | Taxa adicional sobre o valor parcelado. |

Fonte: Elaborado pelos autores.

A classe “Tanatopraxia” registra os procedimentos técnicos de conservação e preparação do corpo realizados pelas funerárias. Contém dados como data, descrição do serviço e condição do corpo no momento do atendimento. Esses registros são fundamentais tanto para fins operacionais quanto legais, garantindo que os processos sigam protocolos adequados e que haja histórico de cada procedimento realizado.

Quadro 17 – Descrição Classe Tanatopraxia

| Atributo | Tipo | Descrição |
|---------------|----------|---|
| id | Integer | Integer Código único da tanatopraxia. |
| data | DateTime | Data em que a tanatopraxia foi realizada. |
| descricao | String | Descrição do procedimento realizado. |
| condicaoCorpo | double | Condição do corpo no momento da realização do procedimento. |

Fonte: Elaborado pelos autores.

O Enum “RelacaoDependente” define os possíveis vínculos familiares dos dependentes em relação ao titular do plano funerário. Esse recurso padroniza e facilita a classificação e consulta dos dependentes nos contratos, promovendo clareza e organização das informações familiares.

Quadro 18 – Descrição Enum RelacaoDependente

| Atributo | Tipo | Descrição |
|------------|---------|-----------------|
| paiMae | Integer | Pai ou Mãe. |
| filhoFilha | Integer | Filho ou Filha. |
| irmaoIrma | Integer | Irmão ou Irmã. |
| avoAvo | Integer | Avô ou Avó. |

| | | |
|------------------|---------|-----------------------|
| netoNeta | Integer | Neto ou Neta. |
| tioTia | Integer | Tio ou Tia. |
| sobrinhoSobrinha | Integer | Sobrinho ou Sobrinha. |
| primoPrima | Integer | Primo ou Prima. |
| conjugue | Integer | Cônjuge. |
| sogroSogra | Integer | Sogro ou Sogra. |
| genroNora | Integer | Genro ou Nora. |

Fonte: Elaborado pelos autores.

O Enum “FormaPagamento” define os métodos de pagamento aceitos pelo sistema funerário, como débito, crédito, dinheiro e Pix. Ele assegura uma padronização dos registros financeiros e oferece flexibilidade na forma como os clientes escolhem pagar pelos serviços.

Quadro 19 – Descrição Enum FormaPagamento

| Atributo | Tipo | Descrição |
|----------|---------|------------------------|
| Débito | Integer | Pagamento com débito. |
| Crédito | Integer | Pagamento com crédito. |
| Dinheiro | Integer | Pagamento em dinheiro. |
| Pix | Integer | Pagamento com Pix. |

Fonte: Elaborado pelos autores.

A classe “beneficiosContrato” é uma classe fazer a ligação de N para N de planos funerários com benefícios.

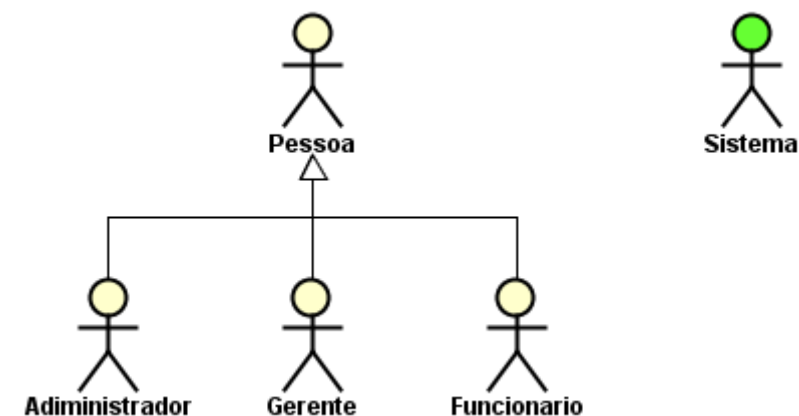
Quadro 20 – Descrição Classe BeneficiosContrato

| Atributo | Tipo | Descrição |
|----------|------|-----------|
|----------|------|-----------|

Fonte: Elaborado pelos autores.

3.3 DEFINIÇÃO DOS ATORES

Figura 5 – Diagrama de Atores



Fonte: Elaborado pelos autores.

No sistema, o ator "Pessoa" abrange todas as interações possíveis, como as operações administrativas realizadas pelos administradores, gerenciando as atividades das funerárias em geral, o gerente faz as operações dentro da funerária e o de registro conduzidas pelos funcionários. Dessa forma, "Pessoa" reflete a totalidade dos usuários que utilizam o sistema em diferentes níveis e com diferentes permissões, garantindo que todas as funcionalidades possam ser associadas a um único ator.

3.4 LISTA CASO DE USO

Guedes (2018) destaca que os casos de uso visam identificar tanto os usuários do sistema quanto os requisitos necessários, definindo "serviços, tarefas ou funcionalidades identificados como necessários ao software e que podem ser utilizados de alguma maneira pelos atores que interagem com o sistema". Para garantir uma compreensão completa do funcionamento do sistema, foram elaboradas duas listas. A primeira apresenta todas as mensagens que o sistema poderá retornar (Quadro 21) e a segunda, os requisitos de casos de uso (Quadro 22).

A lista de mensagens são o tipo de resposta que o sistema retornará quando um caso de uso for executado, quando o usuário fizer alguma interação com as funcionalidades do sistema.

Quadro 21 – Lista de mensagens

| Nº da Mensagem | Conteúdo da Mensagem | Descrição |
|----------------|---------------------------------|--|
| Msg1 | Cadastrado com Sucesso. | Mensagem ao cadastrar com um registro sucesso. |
| Msg2 | Alterado com Sucesso. | Mensagem ao alterar com um registro sucesso. |
| Msg3 | Inativado com Sucesso. | Mensagem ao inativar um registro sucesso. |
| Msg4 | Login realizado. | Mensagem de login realizado com sucesso. |
| Msg5 | Falha ao Cadastrar. | Mensagem de erro ao cadastrar. |
| Msg6 | Falha ao Alterar. | Mensagem de erro ao editar. |
| Msg7 | Erro ao Conectar ao Banco | Mensagem de erro ao tentar conectar com o banco. |
| Msg8 | Dados já existentes no sistema. | Mensagem de dados duplicados no sistema. |
| Msg9 | Retorna Dados | Retorna os dados. |

Fonte: Elaborado pelos autores.

A lista de casos de uso mostra as informações sobre cada caso de uso do sistema e a forma como os atores interagem com ele. O Quadro 22 apresenta a lista completa dos casos de uso do sistema.

Quadro 22 – Lista de casos de uso

| Nº | Descrição do Caso de Uso | Entrada | Caso de Uso | Resposta |
|----|--------------------------------|--|-------------------|----------|
| 1 | Administrador faz login | Informa dados de acesso como email e senha | Login | Msg4 |
| 2 | Administrador Cadastra Usuário | Informa dados do novo usuário | Cadastrar Usuário | Msg1 |

| | | | | |
|----|---|---|-----------------------------|------|
| 3 | Administrador Cadastra Empresa | Informa dados da empresa | Cadastrar Empresa | Msg1 |
| 4 | Administrador Cadastra Plano Funerário | Informa dados do plano funerário | Cadastrar Planos Funerários | Msg1 |
| 5 | Administrador Cadastra Benefícios | Informa dados do benefício | Cadastrar Benefício | Msg1 |
| 6 | Administrador Cadastra Dependente | Informa dados do dependente | Cadastrar Dependente | Msg1 |
| 7 | Administrador Cadastra Contrato | Informa dados do contrato | Cadastrar Contrato | Msg1 |
| 8 | Administrador Cadastra Parcelamento | Informa dados do parcelamento | Cadastrar Parcelamento | Msg1 |
| 9 | Administrador Cadastra Cliente | Informa dados do cliente | Cadastrar Cliente | Msg1 |
| 10 | Administrador Cadastra Pessoa Falecida | Informa dados da pessoa falecida | Cadastrar Pessoa Falecida | Msg1 |
| 11 | Administrador Cadastra Velório | Informa dados do velório | Cadastrar Velório | Msg1 |
| 12 | Administrador Cadastra Cremações | Informa dados da cremação | Cadastrar Cremações | Msg1 |
| 13 | Administrador Cadastra Tanatopraxia | Informa dados da operação de tanatopraxia | Cadastrar Tanatopraxia | Msg1 |
| 14 | Administrador Cadastra Transporte | Informa dados de transporte | Cadastrar Transporte | Msg1 |
| 15 | Administrador Altera Usuário | Informa dados do usuário | Altera Usuário | Msg2 |

| | | | | |
|----|---|---|--------------------------|------|
| 16 | Administrador Altera Empresa | Informa dados da empresa | Altera Empresa | Msg2 |
| 17 | Administrador Altera Plano Funerário | Informa dados do plano funerário | Altera Planos Funerários | Msg2 |
| 18 | Administrador Altera Benefícios | Informa dados do benefício | Altera Benefício | Msg2 |
| 19 | Administrador Altera Dependente | Informa dados do dependente | Altera Dependente | Msg2 |
| 20 | Administrador Altera Contrato | Informa dados do contrato | Altera Contrato | Msg2 |
| 21 | Administrador Altera Parcelamento | Informa dados do parcelamento | Altera Parcelamento | Msg2 |
| 22 | Administrador Altera Cliente | Informa dados do cliente | Altera Cliente | Msg2 |
| 23 | Administrador Altera Pessoa Falecida | Informa dados da pessoa falecida | Altera Pessoa Falecida | Msg2 |
| 24 | Administrador Altera Velório | Informa dados do velório | Altera Velório | Msg2 |
| 25 | Administrador Altera Cremações | Informa dados da cremação | Altera Cremações | Msg2 |
| 26 | Administrador Altera Tanatopraxia | Informa dados da operação de tanatopraxia | Altera Tanatopraxia | Msg2 |
| 27 | Administrador Altera Transporte | Informa dados de transporte | Altera Transporte | Msg2 |
| 28 | Administrador Inativa Usuário | Id usuário | Inativar Usuário | Msg3 |
| 29 | Administrador Inativa Empresa | Id empresa | Inativar Empresa | Msg3 |

| | | | | |
|----|---|--------------------|-------------------------------|------|
| 30 | Administrador Inativa Plano Funerário | Id plano funerário | Inativar Planos Funerários | Msg3 |
| 31 | Administrador Inativa Benefícios | Id benefício | Inativar Benefício | Msg3 |
| 32 | Administrador Inativa Dependente | Id dependente | Inativar Dependente | Msg3 |
| 33 | Administrador Inativa Contrato | Id contrato | Inativar Contrato | Msg3 |
| 34 | Administrador Inativa Cliente | Id cliente | Inativar Cliente | Msg3 |
| 35 | Administrador solicita lista de Usuários | Id usuário | Listar Usuário | Msg9 |
| 36 | Administrador solicita lista de Empresas | Id empresa | Listar Empresa | Msg9 |
| 37 | Administrador solicita lista de Planos Funerários | Id plano funerário | Listar Planos Funerários | Msg9 |
| 38 | Administrador solicita lista de Benefícios | Id benefício | Listar Benefício | Msg9 |
| 39 | Administrador solicita lista de Dependentes | Id dependente | Listar Dependente | Msg9 |
| 40 | Administrador solicita lista de Contratos | Id contrato | Listar Contrato | Msg9 |
| 41 | Administrador solicita lista de Parcelamentos | Id parcelamento | Listar Parcelamento | Msg9 |

| | | | | |
|----|---|--|-----------------------------|------|
| 42 | Administrador solicita lista de Clientes | Id cliente | Listar Cliente | Msg9 |
| 43 | Administrador solicita lista de Pessoas Falecidas | Id pessoa falecida | Listar Pessoa Falecida | Msg9 |
| 44 | Administrador solicita lista de Velórios | Id velório | Listar Velório | Msg9 |
| 45 | Administrador solicita lista de Cremações | Id cremação | Listar Cremações | Msg9 |
| 46 | Administrador solicita lista Tanatopraxias | Id tanatopraxia | Listar Tanatopraxia | Msg9 |
| 47 | Administrador solicita lista Transportes | Id transporte | Listar Transporte | Msg9 |
| 48 | Gerente faz login | Informa dados de acesso como email e senha | Login | Msg4 |
| 49 | Gerente Cadastra Plano Funerário | Informa dados do plano funerário | Cadastrar Planos Funerários | Msg1 |
| 50 | Gerente Cadastra Benefícios | Informa dados do benefício | Cadastrar Benefício | Msg1 |
| 51 | Gerente Cadastra Dependente | Informa dados do dependente | Cadastrar Dependente | Msg1 |
| 52 | Gerente Cadastra Contrato | Informa dados do contrato | Cadastrar Contrato | Msg1 |
| 53 | Gerente Cadastra Parcelamento | Informa dados do parcelamento | Cadastrar Parcelamento | Msg1 |
| 54 | Gerente Cadastra Cliente | Informa dados do cliente | Cadastrar Cliente | Msg1 |
| 55 | Gerente Cadastra Pessoa Falecida | Informa dados da pessoa falecida | Cadastrar Pessoa Falecida | Msg1 |

| | | | | |
|----|--------------------------------|---|---------------------------|------|
| 56 | Gerente Cadastra Velório | Informa dados do velório | Cadastrar Velório | Msg1 |
| 57 | Gerente Cadastra Cremações | Informa dados da cremação | Cadastrar Cremações | Msg1 |
| 58 | Gerente Cadastra Tanatopraxia | Informa dados da operação de tanatopraxia | Cadastrar Tanatopraxia | Msg1 |
| 59 | Gerente Cadastra Transporte | Informa dados de transporte | Cadastrar Transporte | Msg1 |
| 60 | Gerente Altera Empresa | Informa dados da sua empresa | Alterar Empresa | Msg2 |
| 61 | Gerente Altera Plano Funerário | Informa dados do plano funerário | Alterar Planos Funerários | Msg2 |
| 62 | Gerente Altera Benefícios | Informa dados do benefício | Alterar Benefício | Msg2 |
| 63 | Gerente Altera Dependente | Informa dados do dependente | Alterar Dependente | Msg2 |
| 64 | Gerente Altera Contrato | Informa dados do contrato | Alterar Contrato | Msg2 |
| 65 | Gerente Altera Parcelamento | Informa dados do parcelamento | Alterar Parcelamento | Msg2 |
| 66 | Gerente Altera Cliente | Informa dados do cliente | Alterar Cliente | Msg2 |
| 67 | Gerente Altera Pessoa Falecida | Informa dados da pessoa falecida | Alterar Pessoa Falecida | Msg2 |
| 68 | Gerente Altera Velório | Informa dados do velório | Alterar Velório | Msg2 |
| 69 | Gerente Altera Cremações | Informa dados da cremação | Alterar Cremações | Msg2 |
| 70 | Gerente Altera Tanatopraxia | Informa dados da operação de tanatopraxia | Alterar Tanatopraxia | Msg2 |

| | | | | |
|----|---|-----------------------------|----------------------------|------|
| 71 | Gerente Altera Transporte | Informa dados de transporte | Altera Transporte | Msg2 |
| 72 | Gerente Inativa Plano Funerário | Id plano funerário | Inativar Planos Funerários | Msg3 |
| 73 | Gerente inativa Usuário | Id usuário | Inativar Usuário | Msg3 |
| 74 | Gerente Inativa Benefícios | Id benefício | Inativar Benefício | Msg3 |
| 75 | Gerente Inativa Dependente | Id dependente | Inativar Dependente | Msg3 |
| 76 | Gerente Inativa Contrato | Id contrato | Inativar Contrato | Msg3 |
| 77 | Gerente Inativa Cliente | Id cliente | Inativar Cliente | Msg3 |
| 78 | Gerente solicita lista de Usuários da empresa | Id usuário | Listar Usuário | Msg9 |
| 79 | Gerente solicita lista de Planos Funerários | Id plano funerário | Listar Planos Funerários | Msg9 |
| 80 | Gerente solicita lista de Benefícios | Id benefício | Listar Benefício | Msg9 |
| 81 | Gerente solicita lista de Dependentes | Id dependente | Listar Dependente | Msg9 |
| 82 | Gerente solicita lista de Contratos | Id contrato | Listar Contrato | Msg9 |
| 83 | Gerente solicita lista de Parcelamentos | Id parcelamento | Listar Parcelamento | Msg9 |
| 84 | Gerente solicita lista de Clientes | Id cliente | Listar Cliente | Msg9 |

| | | | | |
|----|---|--|-----------------------------|------|
| 85 | Gerente solicita lista de Pessoas Falecidas | Id pessoa falecida | Listar Pessoa Falecida | Msg9 |
| 86 | Gerente solicita lista de Velórios | Id velório | Listar Velório | Msg9 |
| 87 | Gerente solicita lista de Cremações | Id cremação | Listar Cremações | Msg9 |
| 88 | Gerente solicita lista Tanatopraxias | Id tanatopraxia | Listar Tanatopraxia | Msg9 |
| 89 | Gerente solicita lista Transportes | Id transporte | Listar Transporte | Msg9 |
| 90 | Funcionário faz login | Informa dados de acesso como email e senha | Login | Msg4 |
| 91 | Funcionário Cadastra Plano Funerário | Informa dados do plano funerário | Cadastrar Planos Funerários | Msg1 |
| 92 | Funcionário Cadastra Benefícios | Informa dados do benefício | Cadastrar Benefício | Msg1 |
| 93 | Funcionário Cadastra Dependente | Informa dados do dependente | Cadastrar Dependente | Msg1 |
| 94 | Funcionário Cadastra Contrato | Informa dados do contrato | Cadastrar Contrato | Msg1 |
| 95 | Funcionário Cadastra Parcelamento | Informa dados do parcelamento | Cadastrar Parcelamento | Msg1 |
| 96 | Funcionário Cadastra Cliente | Informa dados do cliente | Cadastrar Cliente | Msg1 |
| 97 | Funcionário Cadastra Pessoa Falecida | Informa dados da pessoa falecida | Cadastrar Pessoa Falecida | Msg1 |

| | | | | |
|-----|--|---|-----------------------------|------|
| 98 | Funcionário Cadastra Velório | Informa dados do velório | Cadastrar Velório | Msg1 |
| 99 | Funcionário Cadastra Cremações | Informa dados da cremação | Cadastrar Cremações | Msg1 |
| 100 | Funcionário Cadastra Tanatopraxia | Informa dados da operação de tanatopraxia | Cadastrar Tanatopraxia | Msg1 |
| 101 | Funcionário Cadastra Transporte | Informa dados de transporte | Cadastrar Transporte | Msg1 |
| 102 | Funcionário Altera Plano Funerário | Informa dados do plano funerário | Altera Planos Funerários | Msg2 |
| 103 | Funcionário Altera Benefícios | Informa dados do benefício | Altera Benefício | Msg2 |
| 104 | Funcionário Altera Dependente | Informa dados do dependente | Altera Dependente | Msg2 |
| 105 | Funcionário Altera Contrato | Informa dados do contrato | Altera Contrato | Msg2 |
| 106 | Funcionário Altera Parcelamento | Informa dados do parcelamento | Altera Parcelamento | Msg2 |
| 107 | Funcionário Altera Cliente | Informa dados do cliente | Altera Cliente | Msg2 |
| 108 | Funcionário Altera Pessoa Falecida | Informa dados da pessoa falecida | Altera Pessoa Falecida | Msg2 |
| 109 | Funcionário Altera Velório | Informa dados do velório | Altera Velório | Msg2 |
| 110 | Funcionário Altera Cremações | Informa dados da cremação | Altera Cremações | Msg2 |

| | | | | |
|-----|--|---|-------------------------------|------|
| 111 | Funcionário Altera Tanatopraxia | Informa dados da operação de tanatopraxia | Altera Tanatopraxia | Msg2 |
| 112 | Funcionário Altera Transporte | Informa dados de transporte | Altera Transporte | Msg2 |
| 113 | Funcionário Inativa Plano Funerário | Id plano funerário | Inativar Planos Funerários | Msg3 |
| 114 | Funcionário Inativa Benefícios | Id benefício | Inativar Benefício | Msg3 |
| 115 | Funcionário Inativa Dependente | Id dependente | Inativar Dependente | Msg3 |
| 116 | Funcionário Inativa Contrato | Id contrato | Inativar Contrato | Msg3 |
| 117 | Funcionário Inativa Cliente | Id cliente | Inativar Cliente | Msg3 |
| 118 | Funcionário solicita lista de Usuários da empresa | Id usuário | Listar Usuário | Msg9 |
| 119 | Funcionário solicita lista de Planos Funerários | Id plano funerário | Listar Planos Funerários | Msg9 |
| 120 | Funcionário solicita lista de Benefícios | Id benefício | Listar Benefício | Msg9 |
| 121 | Funcionário solicita lista de Dependentes | Id dependente | Listar Dependente | Msg9 |
| 123 | Funcionário solicita lista de Contratos | Id contrato | Listar Contrato | Msg9 |

| | | | | |
|-----|---|--------------------|-------------------------------|------|
| 124 | Funcionário solicita lista de Parcelamentos | Id parcelamento | Listar Parcelamento | Msg9 |
| 125 | Funcionário solicita lista de Clientes | Id cliente | Listar Cliente | Msg9 |
| 126 | Funcionário solicita lista de Pessoas Falecidas | Id pessoa falecida | Listar Pessoa Falecida | Msg9 |
| 127 | Funcionário solicita lista de Velórios | Id velório | Listar Velório | Msg9 |
| 128 | Funcionário solicita lista de Cremações | Id cremação | Listar Cremações | Msg9 |
| 129 | Funcionário solicita lista Tanatopraxias | Id tanatopraxia | Listar Tanatopraxia | Msg9 |
| 130 | Funcionário solicita lista Transportes | Id transporte | Listar Transporte | Msg9 |
| 131 | Gerente Lista Usuário pela sua empresa | Id empresa | Pesquisa Usuário pela Empresa | Msg9 |
| 132 | Gerente se cadastra no sistema | Informa seus dados | Cadastrar Usuário | Msg1 |
| 133 | Funcionário se cadastra no sistema | Informa seus dados | Cadastrar Usuário | Msg1 |
| 134 | Administrador Lista Usuário pela sua empresa | Id empresa | Pesquisa Usuário pela Empresa | Msg9 |

Fonte: Elaborado pelos autores.

3.5 DIAGRAMA DE CASO DE USO

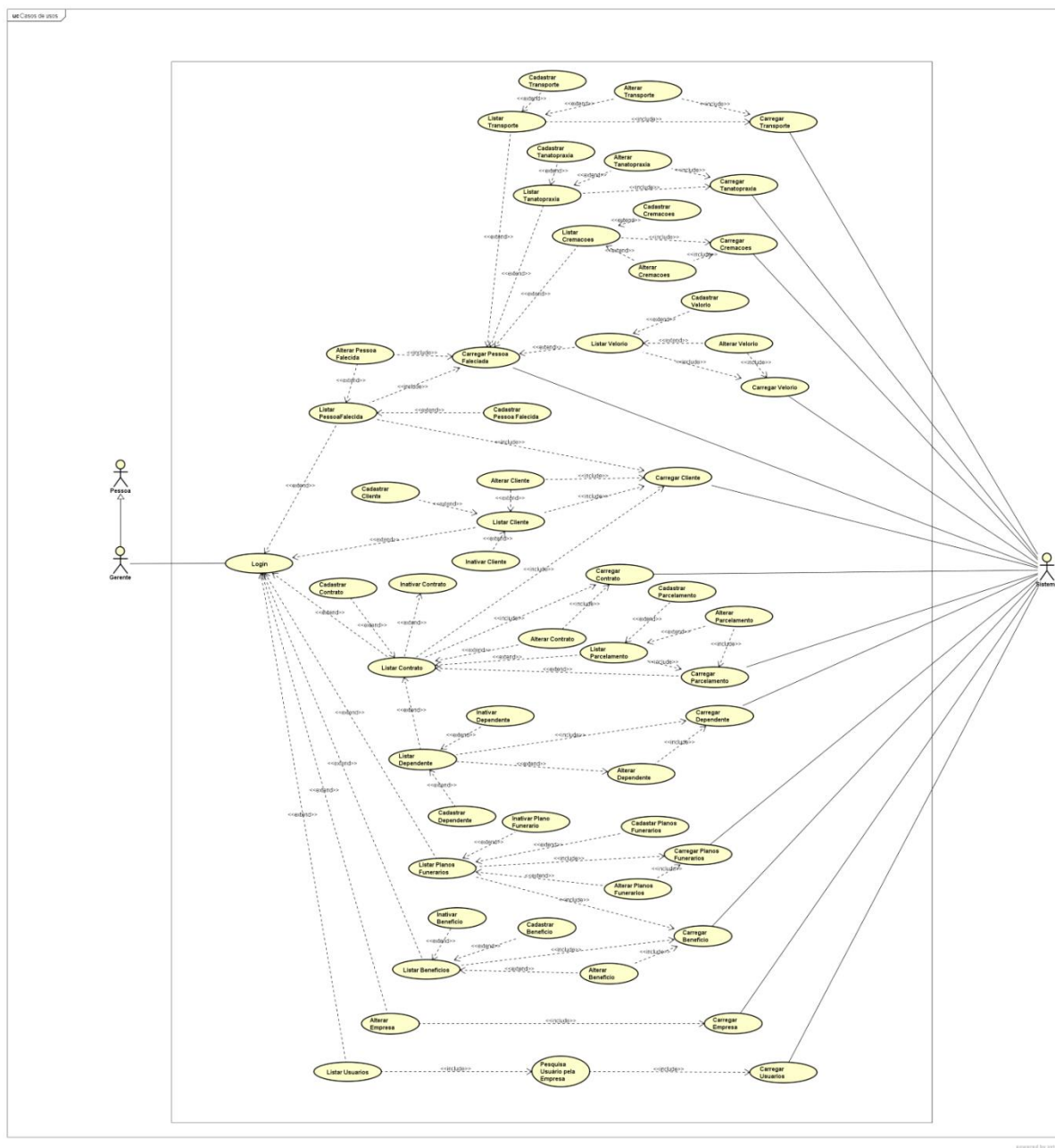
Guedes (2018) descreve o diagrama de casos de uso como uma representação do sistema a partir da perspectiva do usuário, cujo objetivo é apresentar de forma geral as funcionalidades que o sistema deve oferecer ao usuário. Além disso, Guedes (2018) destaca que esse diagrama é uma ferramenta essencial para identificar e compreender os requisitos do sistema, auxiliando na especificação, visualização e documentação das características, funções e serviços que o sistema deve proporcionar.

O Diagrama de Casos de Uso Geral é uma representação visual que oferece uma visão **abrangente** das interações entre todos os atores e o sistema como um todo. As funcionalidades que o sistema deve proporcionar, sem se aprofundar na lógica interna ou na ordem de execução dessas ações especificando o escopo do sistema.

Na Figura 6 apresenta o diagrama de casos de uso sob a perspectiva do “Administrador”, o qual ocupa o nível hierárquico mais alto no sistema. O administrador é responsável pela gestão completa das informações e configurações da plataforma, podendo realizar ações como cadastrar, alterar, excluir, listar e carregar dados de usuários, contratos, clientes, planos, entre outros, além de acompanhar painéis de indicadores e relatórios gerenciais. Sua função centraliza o controle do sistema, incluindo a administração dos demais usuários e definição de permissões, o que reflete sua responsabilidade por assegurar o funcionamento pleno e seguro da aplicação.

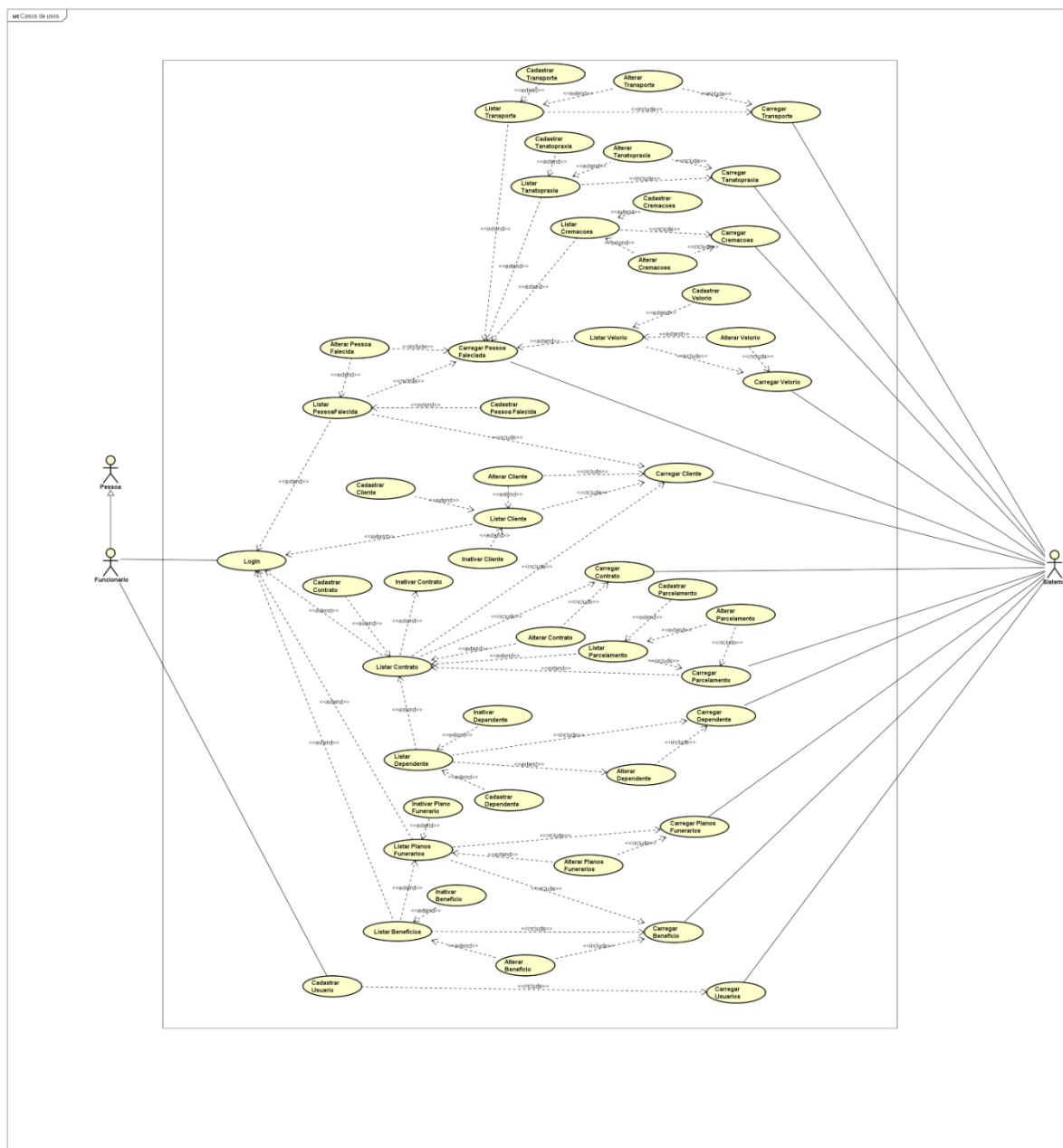
Já as imagens 7 e 8 ilustram os perfis de “Gerente” e “Funcionário”, que possuem níveis de acesso distintos e mais restritos em comparação ao Administrador. O Gerente tem acesso ampliado em relação ao Funcionário, podendo realizar operações gerenciais e acompanhar métricas do sistema, mas com limitações no que diz respeito à gestão de usuários e configurações críticas. Por sua vez, o Funcionário possui um conjunto de permissões ainda mais restrito, podendo apenas consultar, editar e carregar algumas informações no sistema.

Figura 7 – Diagrama de Caso de Uso Geral - Visão Gerente



Fonte: Elaborado pelos autores.

Figura 8 – Diagrama de Caso de Uso Geral – Visão Funcionário



Fonte: Elaborado pelos autores.

3.6 DIAGRAMA DE CASO DE USO INDIVIDUAIS

Para Guedes (2011), a modelagem de casos de uso é essencial para a definição dos requisitos do sistema, funcionando como o ponto de partida para a modelagem, pois assegura que as expectativas dos stakeholders sejam atendidas e facilita a comunicação eficaz entre desenvolvedores, analistas e usuários.

| Documentação | |
|---|---|
| Nome do caso de uso | Listar Tanatopraxia |
| Ator principal | Usuário(Administrador) |
| Resumo | Descreve as ações para o usuário listar as tanatopraxias. |
| Pré-condições | Administrador logado e autenticado. |
| Pós-condições | Lista Tanatopraxia. |
| Fluxo Normal | |
| Ações do ator | Ações do sistema |
| 1. Administrador faz requisição de lista de tanatopraxia. | |
| | 2. Sistema exibe lista de Tanatopraxias. |
| Fluxo Alternativo | |
| | 1. Não é possível exibir a lista de Tanatopraxia. |
| | 2. Não possui tanatopraxia cadastrados. |

Fonte: Elaborado pelos autores.

No Quadro 24, referente ao cadastro de tanatopraxia, o fluxo normal inicia quando o “Administrador” solicita o cadastro de um novo registro. O sistema então exibe um formulário para preenchimento dos dados. Após o administrador inserir as informações, o sistema realiza a validação dos dados. Se forem válidos, o sistema cadastra a tanatopraxia no banco de dados e retorna uma mensagem de sucesso, fechando o formulário e exibindo a listagem atualizada. No fluxo alternativo, caso os dados inseridos sejam inválidos, o sistema retorna uma mensagem de erro informando a inconsistência nos dados.

Quadro 24 – Documentação - Cadastrar Tanatopraxia

| Documentação | |
|---------------------|------------------------|
| Nome do caso de uso | Cadastrar Tanatopraxia |
| Ator principal | Usuário(Administrador) |

| | |
|--|--|
| Resumo | Descreve as ações para o usuário cadastrar um novo registro de tanatopraxia. |
| Pré-condições | Administrador logado e autenticado. |
| Pós-condições | Fecha o formulário e lista Tanatopraxia. |
| Fluxo Normal | |
| Ações do ator | Ações do sistema |
| 1. Administrador faz requisição de cadastro de tanatopraxia. | |
| | 2. Sistema exhibe formulário de cadastro de Tanatopraxias. |
| 3. Administrador insere as informações. | |
| | 4. Sistema valida as informações. |
| | 5. Sistema cadastra no banco de dados. |
| | 6. Sistema retorna mensagem de cadastrado com sucesso. |
| Fluxo Alternativo | |
| | 4. Sistema valida as informações. |
| | 5. Sistema retorna mensagem de dados inválidos. |

Fonte: Elaborado pelos autores.

No Quadro 25, referente à alteração de tanatopraxia, o fluxo normal inicia com a solicitação de edição de um registro já existente. O sistema exhibe o formulário de alteração, permitindo ao “Administrador” modificar os dados. Após o preenchimento, o sistema valida e atualiza as informações no banco de dados, retornando uma mensagem de sucesso. No fluxo alternativo, o sistema pode retornar mensagens de erro caso os dados sejam inválidos ou se o registro de tanatopraxia não existir no sistema.

Quadro 25 – Documentação - Alterar Tanatopraxia

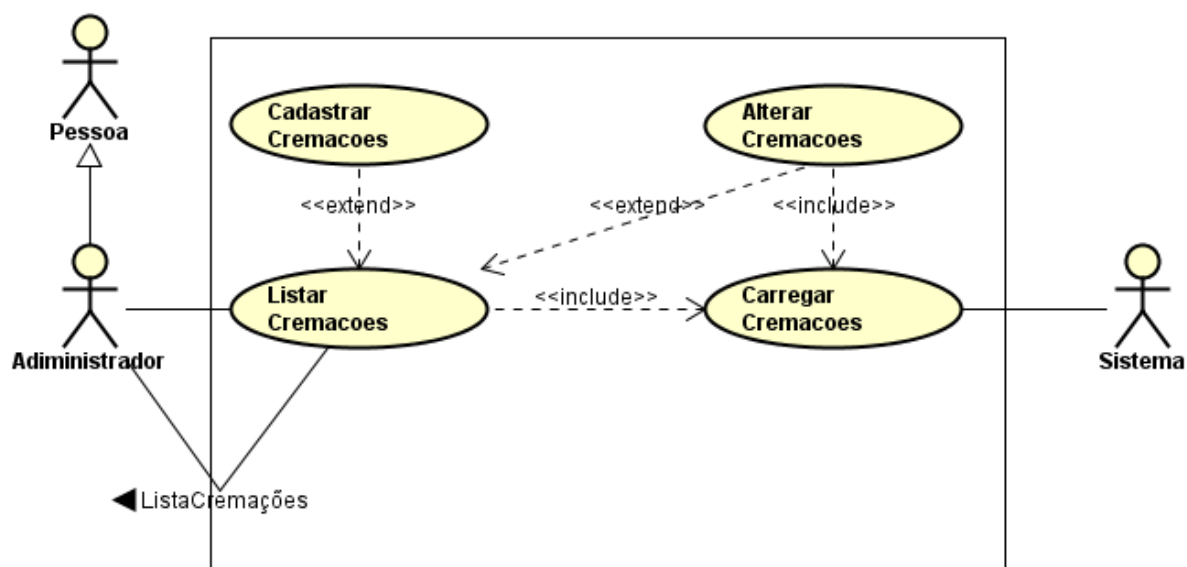
| |
|---------------------|
| Documentação |
|---------------------|

| | |
|---|---|
| Nome do caso de uso | Alterar Tanatopraxia |
| Ator principal | Usuário (Administrador) |
| Resumo | Descreve as ações para o usuário alterar as informações de um registro de tanatopraxia. |
| Pré-condições | Tanatopraxia cadastrada no sistema. |
| Pós-condições | Fecha o formulário e lista Tanatopraxia. |
| Fluxo Normal | |
| Ações do ator | Ações do sistema |
| 1. Administrador faz requisição de alteração de tanatopraxia. | |
| | 2. Sistema exibe formulário de alteração de Tanatopraxias. |
| 3. Administrador insere as informações. | |
| | 4. Sistema valida as informações. |
| | 5. Sistema altera no banco de dados. |
| | 6. Sistema retorna mensagem de alteração foi realizada com sucesso. |
| Fluxo Alternativo | |
| | 4. Sistema valida as informações. |
| | 5. Sistema retorna mensagem de dados inválidos. |
| | 7. Sistema retorna mensagem de tanatopraxia não cadastrada no sistema. |

Fonte: Elaborado pelos autores.

3.6.2 DIAGRAMA DE CASO DE USO INDIVIDUAL – ADMINISTRADOR LISTA CREMAÇÕES

Figura 10 – Diagrama de Caso de Uso Individual – Administrador Lista Cremações



Fonte: Elaborado pelos autores.

No Quadro 26, é descrito o fluxo normal e alternativo para o “Administrador” listar as cremações. Para interagir com o sistema, o “Administrador” deve estar logado e autenticado. No fluxo normal, o “Administrador” solicita a lista de cremações, e o “Sistema” retorna a listagem de cremações para a visualização. No fluxo alternativo, se ocorrer um erro na requisição da lista, o sistema retorna uma mensagem de erro.

Quadro 26 – Documentação - Listar Cremação

| Documentação | |
|---------------------|---|
| Nome do caso de uso | Listar Cremação |
| Ator principal | Usuário (Administrador) |
| Resumo | Descreve as ações para o usuário listar as Cremações. |

| | |
|---|--|
| Pré-condições | Administrador logado e autenticado. |
| Pós-condições | Lista Cremação. |
| Fluxo Normal | |
| Ações do ator | Ações do sistema |
| 3. Administrador faz requisição de lista de cremação. | |
| | 4. Sistema exibe lista de Cremações. |
| Fluxo Alternativo | |
| | 3. Não é possível exibir a lista de Cremações. |
| | 4. Não possui cremações cadastrados. |

Fonte: Elaborado pelos autores.

No Quadro 27, demonstra as etapas para realizar o cadastro de cremação pelo “Administrador”. O fluxo normal inicia com a solicitação do formulário de registro, seguida pelo preenchimento e envio dos dados. O sistema valida as informações e, se estiverem corretas, efetua o cadastro com sucesso. No fluxo alternativo, quando os dados não são válidos, o sistema informa o erro.

Quadro 27 – Documentação - Cadastrar Cremação

| Documentação | |
|---------------------|--|
| Nome do caso de uso | Cadastrar Cremação |
| Ator principal | Usuário (Administrador) |
| Resumo | Descreve as ações para o usuário cadastrar um novo registro de cremação. |
| Pré-condições | Administrador logado e autenticado. |
| Pós-condições | Fecha o formulário e lista Cremação. |
| Fluxo Normal | |
| Ações do ator | Ações do sistema |

| | |
|--|---|
| 7. Administrador faz requisição de cadastro de cremação. | |
| | 8. Sistema exibe formulário de cadastro de Cremações. |
| 9. Administrador insere as informações. | |
| | 10. Sistema valida as informações. |
| | 11. Sistema cadastra no banco de dados. |
| | 12. Sistema retorna mensagem de cadastrado com sucesso. |
| Fluxo Alternativo | |
| | 4. Sistema valida as informações. |
| | 5. Sistema retorna mensagem de dados inválidos. |

Fonte: Elaborado pelos autores.

No Quadro 28, demonstra a alteração de cremação, o fluxo normal inicia com a solicitação de edição de um registro já existente. O sistema exibe o formulário de alteração e permite que o “Administrador” modifique os dados necessários. Após o preenchimento, o sistema valida as informações e atualiza o registro no banco de dados, retornando uma mensagem de sucesso. No fluxo alternativo, o sistema pode apresentar mensagens de erro caso os dados sejam inválidos ou se o registro não estiver cadastrado.

Quadro 28 – Documentação - Alterar Cremação

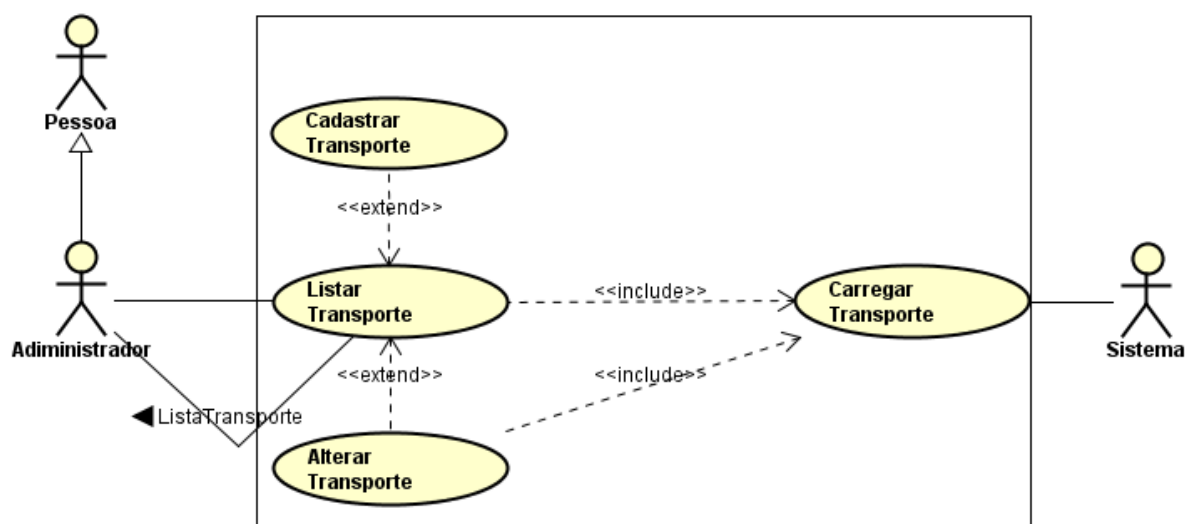
| Documentação | |
|---------------------|---|
| Nome do caso de uso | Alterar Cremação |
| Ator principal | Usuário (Administrador) |
| Resumo | Descreve as ações para o usuário alterar as informações de um registro de cremação. |
| Pré-condições | Cremação cadastrada no sistema. |
| Pós-condições | Fecha o formulário e lista Cremação. |

| Fluxo Normal | |
|---|--|
| Ações do ator | Ações do sistema |
| 8. Administrador faz requisição de alteração de cremação. | |
| | 9. Sistema exibe formulário de alteração de Cremação. |
| 10. Administrador insere as informações. | |
| | 11. Sistema valida as informações. |
| | 12. Sistema altera no banco de dados. |
| | 13. Sistema retorna mensagem de alteração foi realizada com sucesso. |
| Fluxo Alternativo | |
| | 4. Sistema valida as informações. |
| | 5. Sistema retorna mensagem de dados inválidos. |
| | 14. Sistema retorna mensagem de cremação não cadastrada no sistema. |

Fonte: Elaborado pelos autores.

3.6.3 DIAGRAMA DE CASO DE USO INDIVIDUAL – ADMINISTRADOR LISTA TRANSPORTE

Figura 11 – Diagrama de Caso de Uso Individual – Administrador Lista Transporte



Fonte: Elaborado pelos autores.

No Quadro 29, é descrito o fluxo normal e alternativo para o “Administrador” listar os transportes. Para interagir com o sistema, o “Administrador” deve estar logado e autenticado. No fluxo normal, o “Administrador” solicita a lista de transportes, e o “Sistema” retorna a listagem de transportes para a visualização. No fluxo alternativo, se ocorrer um erro na requisição da lista, o sistema retorna uma mensagem de erro.

Quadro 29 – Documentação - Listar Transporte

| Documentação | |
|---------------------|---|
| Nome do caso de uso | Listar Transporte |
| Ator principal | Usuário (Administrador) |
| Resumo | Descreve as ações para o usuário listar os transportes. |
| Pré-condições | Administrador logado e autenticado. |
| Pós-condições | Lista Transporte. |

| Fluxo Normal | |
|---|--|
| Ações do ator | Ações do sistema |
| 5. Administrador faz requisição de lista de transporte. | |
| | 6. Sistema exibe lista de transportes. |
| Fluxo Alternativo | |
| | 5. Não é possível exibir a lista de transportes. |
| | 6. Não possui transportes cadastrados. |

Fonte: Elaborado pelos autores.

No Quadro 30, demonstra as etapas para realizar o cadastro de transporte pelo “Administrador”. O fluxo normal inicia com a solicitação do formulário de registro, seguida pelo preenchimento e envio dos dados. O sistema valida as informações e, se estiverem corretas, efetua o cadastro com sucesso. No fluxo alternativo, quando os dados não são válidos, o sistema informa o erro.

Quadro 30 – Documentação - Cadastrar Transporte

| Documentação | |
|---|--|
| Nome do caso de uso | Cadastrar Transporte |
| Ator principal | Usuário (Administrador) |
| Resumo | Descreve as ações para o usuário cadastrar um novo registro de transporte. |
| Pré-condições | Administrador logado e autenticado. |
| Pós-condições | Fecha o formulário e lista transporte. |
| Fluxo Normal | |
| Ações do ator | Ações do sistema |
| 13. Administrador faz requisição de cadastro de transporte. | |

| | |
|--|---|
| | 14. Sistema exibe formulário de cadastro de transporte. |
| 15. Administrador insere as informações. | |
| | 16. Sistema valida as informações. |
| | 17. Sistema cadastra no banco de dados. |
| | 18. Sistema retorna mensagem de cadastrado com sucesso. |
| Fluxo Alternativo | |
| | 4. Sistema valida as informações. |
| | 5. Sistema retorna mensagem de dados inválidos. |

Fonte: Elaborado pelos autores.

No Quadro 31, demonstra a alteração de Transporte, o fluxo normal inicia com a solicitação de edição de um registro já existente. O sistema exibe o formulário de alteração e permite que o “Administrador” modifique os dados necessários. Após o preenchimento, o sistema valida as informações e atualiza o registro no banco de dados, retornando uma mensagem de sucesso. No fluxo alternativo, o sistema pode apresentar mensagens de erro caso os dados sejam inválidos ou se o registro não estiver cadastrado.

Quadro 31 – Documentação - Alterar Transporte

| Documentação | |
|---------------------|---|
| Nome do caso de uso | Alterar Transporte |
| Ator principal | Usuário (Administrador) |
| Resumo | Descreve as ações para o usuário alterar as informações de um registro de transporte. |
| Pré-condições | Cremação cadastrada no sistema. |
| Pós-condições | Fecha o formulário e lista transporte. |
| Fluxo Normal | |

| Ações do ator | Ações do sistema |
|--|---|
| 15. Administrador faz requisição de alteração de transporte. | |
| | 16. Sistema exibe formulário de alteração de transporte. |
| 17. Administrador insere as informações. | |
| | 18. Sistema valida as informações. |
| | 19. Sistema altera no banco de dados. |
| | 20. Sistema retorna mensagem de alteração foi realizada com sucesso. |
| Fluxo Alternativo | |
| | 4. Sistema valida as informações. |
| | 5. Sistema retorna mensagem de dados inválidos. |
| | 21. Sistema retorna mensagem de transporte não cadastrada no sistema. |

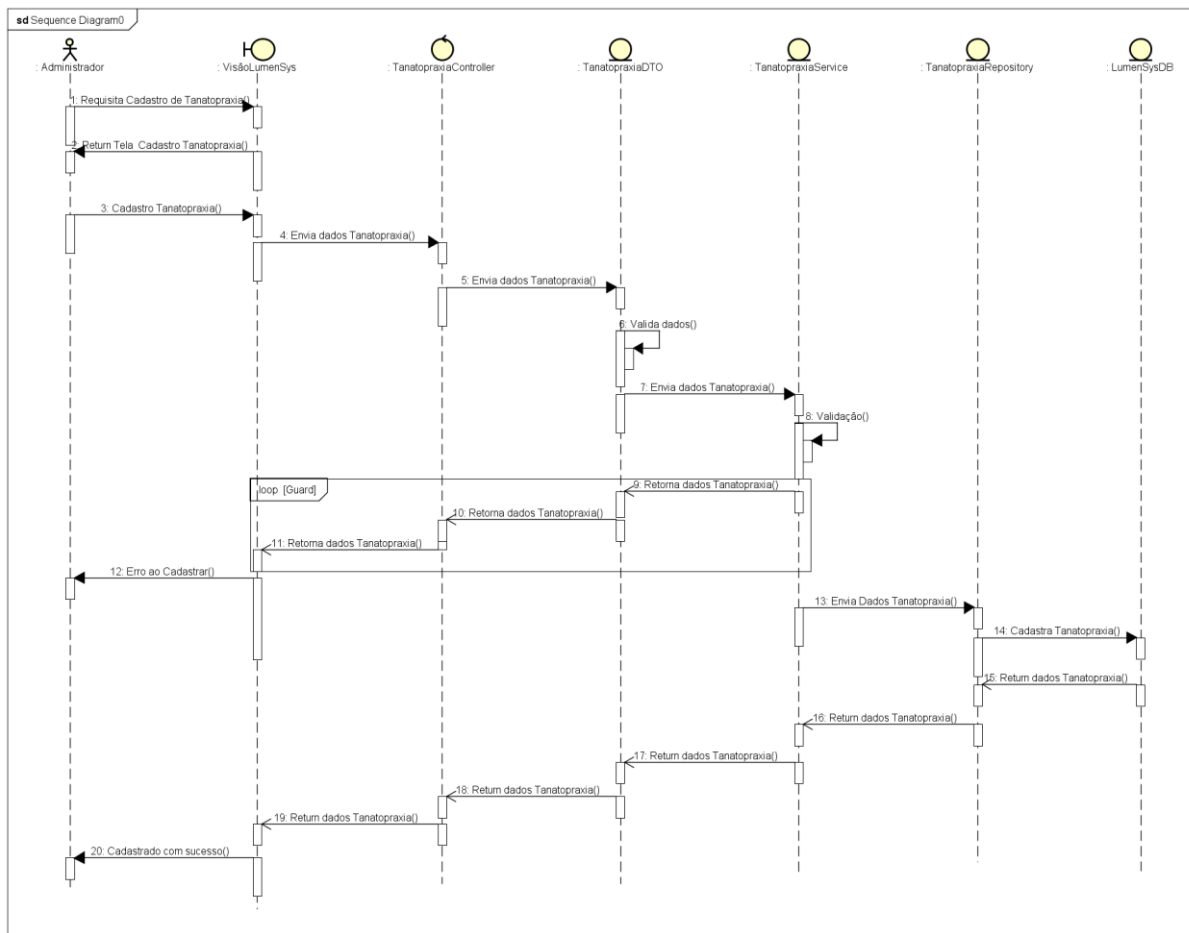
Fonte: Elaborado pelos autores.

3.7 DIAGRAMA DE CASO DE SEQUÊNCIA

O Diagrama de Sequência, conforme Gilleanes Guedes (2018), é um tipo de diagrama comportamental da UML que descreve a ordem temporal das mensagens trocadas entre os objetos durante a execução de um processo geralmente derivado de um caso de uso. Cada diagrama de sequência costuma corresponder a um caso de uso específico, e depende também do diagrama de classes para identificar as classes e instâncias envolvidas

Esses diagramas mostram como os diferentes componentes do sistema interagem ao longo do tempo para executar processos específicos. Com eles, é possível visualizar o fluxo das mensagens trocadas entre usuários e módulos do sistema, facilitando a compreensão das etapas e da ordem das operações realizadas em cada funcionalidade, como mostrado na Figura 11, 12 e 13 para cadastrar, alterar e listar tanatopraxia.

Figura 12 – Diagrama de Sequência – Administrador Cadastra Tanatopraxia



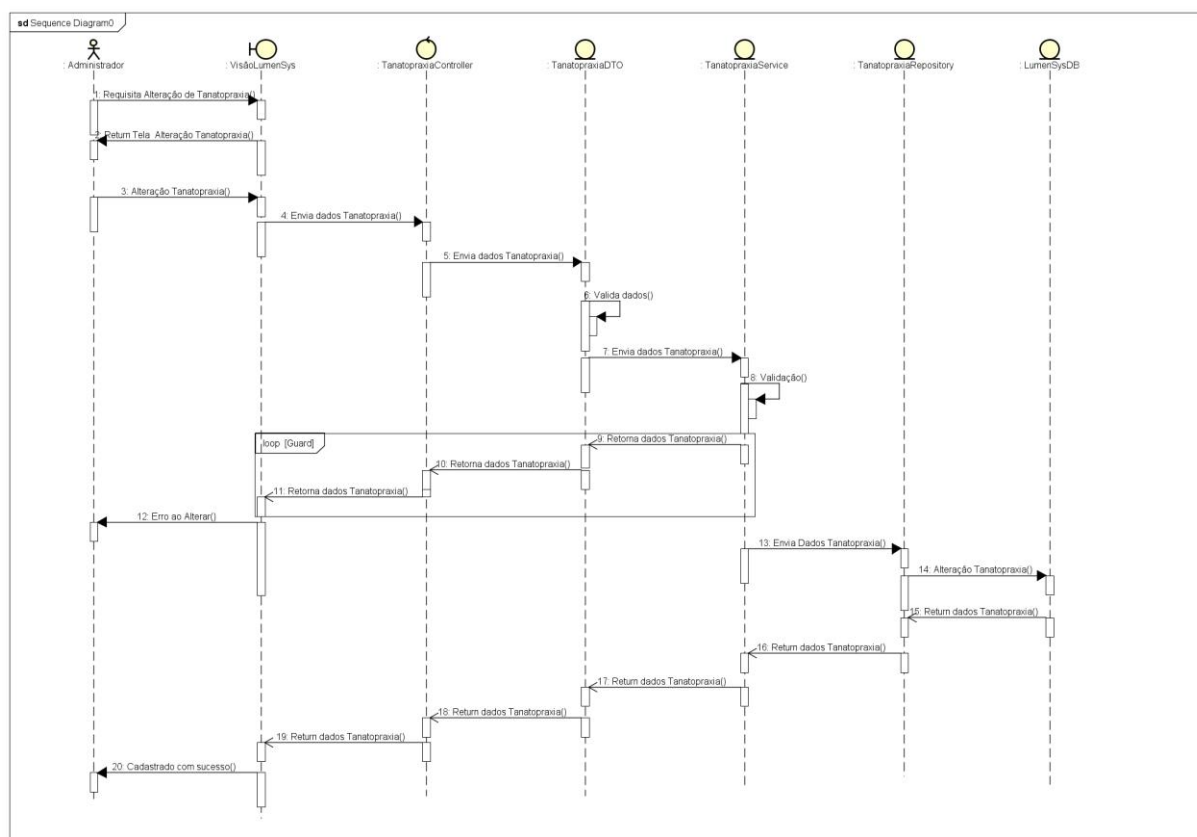
powered by astah®

Fonte: Elaborado pelos autores.

A Figura 12 apresenta o diagrama de sequência do processo de cadastro de uma Tanatopraxia no sistema. O fluxo tem início com a requisição do usuário “Administrador” para abrir o formulário de cadastro. Em seguida, o sistema retorna a tela para o usuário realizar o cadastro. Ao submeter os dados, o fluxo percorre várias camadas do sistema: inicialmente, os dados são enviados ao Controller responsável, que encaminha para a DTO e realiza a validação dos dados. Caso a validação falhe, o sistema retorna um erro ao usuário, interrompendo o processo.

Se a validação for bem-sucedida, os dados são encaminhados para a camada Service, que faz a validação, mas para a regra de negócios, e por fim vez chama o Repositório para persistir as informações no banco de dados. Após a gravação, os dados são retornados em sequência para as camadas superiores até a interface do usuário, que recebe a confirmação de cadastro realizado com sucesso. Dessa forma, o diagrama ilustra as interações entre usuário, interface e os componentes internos do sistema, evidenciando o fluxo de mensagens e a sequência lógica para o cadastro da Tanatopraxia.

Figura 13 – Diagrama de Sequência – Administrador Altera Tanatopraxia



powered by astah

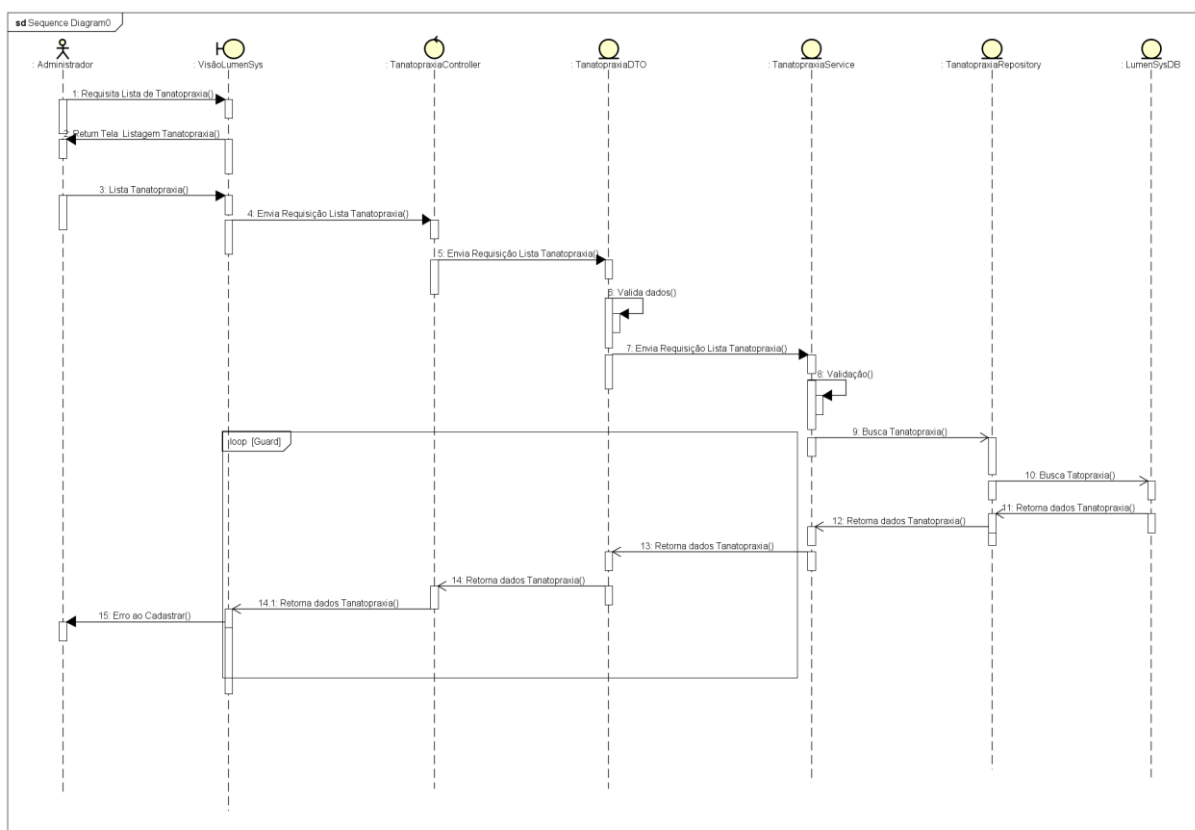
Fonte: Elaborado pelos autores.

A Figura 13 apresenta o diagrama de sequência do processo de alteração de uma Tanatopraxia no sistema. O fluxo se inicia com a solicitação do usuário do tipo Administrador, que requisita a alteração de um registro já existente. O sistema responde carregando a tela correspondente com os dados a serem editados.

Em seguida, o usuário realiza a modificação das informações da Tanatopraxia. Esses dados alterados são enviados inicialmente ao Controller, que os repassa para a DTO, onde ocorre a primeira etapa de validação estrutural. Se os dados estiverem em conformidade, seguem para a camada de serviço, responsável por aplicar as validações de regras de negócio.

Caso alguma falha seja identificada durante essas validações, o fluxo retorna um erro ao usuário, informando a impossibilidade de concluir a operação. Se todas as etapas forem validadas com sucesso, os dados seguem para o Repositório, onde são efetivamente atualizados no banco de dados. Após a alteração no banco, o sistema retorna a resposta pelas camadas superiores até a interface, onde o usuário recebe a confirmação de que o registro foi alterado com sucesso.

Figura 14 – Diagrama de Sequência – Administrador Lista Tanatopraxia



Fonte: Elaborado pelos autores.

A Figura 14 apresenta o diagrama de sequência do processo de listagem de registros de Tanatopraxia no sistema. O fluxo tem início com a requisição do usuário do tipo Administrador, que solicita a visualização da lista de registros. A interface do sistema responde com a exibição da tela de listagem.

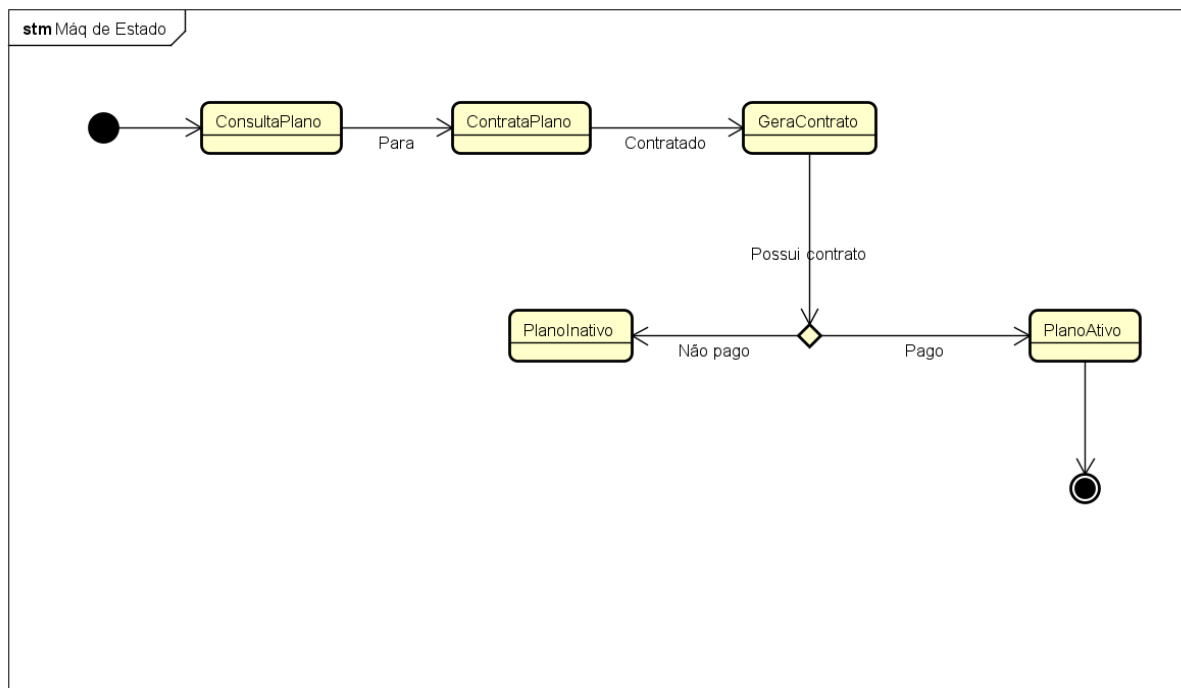
A camada de serviço envia a solicitação para o Repositório, que realiza a consulta no banco de dados. Os dados recuperados são retornados em cadeia pelas camadas superiores, Repositório, Service, DTO e Controller até alcançar a interface de usuário, onde a lista de Tanatopraxia é apresentada, caso alguma falha ocorra durante o processo de consulta, uma mensagem de erro é enviada ao usuário, informando o erro da listagem.

3.8 DIAGRAMA DE MÁQUINA DE ESTADO

Guedes (2018) descreve que o diagrama de máquina de estados ilustra o comportamento de um elemento por meio de um conjunto finito de transições. Este processo é conhecido como máquina de estado comportamental. Assim como o diagrama de sequência, este diagrama pode ser fundamentado em um caso de uso, mas também pode ser empregado para monitorar o estado de outros elementos.

Entre os componentes principais deste diagrama, destacam-se o estado, que representa a condição em que um elemento se encontra durante o processo, e as transições, que correspondem a eventos que provocam mudanças no estado de um elemento, resultando assim em um novo estado (Guedes, 2018).

Figura 15 – Diagrama de Máquina de Estado – Gerenciamento de Contrato



Fonte: Elaborado pelos autores.

Conforme apresentado na Figura 15, o diagrama de máquina de estados do componente “GeraContrato” inicia-se no estado inativo. Para que ocorra a transição para o estado ativo, é necessário que o componente “GeraContrato” mude do estado "PlanoInativo" para "PlanoAtivo". Da mesma forma, a transição do estado ativo para o inativo ocorre quando o componente retorna ao estado "PlanoInativo". Assim, um objeto do sistema pode ser classificado como ativo ou inativo, dependendo do seu estado atual.

4 DEFINIÇÃO DE INTERFACE DO USUÁRIO (UX)

A experiência do usuário, ou UX, refere-se à forma como as pessoas interagem com sistemas, aplicativos ou serviços digitais, e como elas percebem essa interação. UX não está focada apenas no design visual, mas na facilidade, eficiência e satisfação que o usuário sente ao utilizar a interface. O objetivo do UX é criar soluções que facilitem o uso, atendam às expectativas dos usuários e tornem a navegação intuitiva. Isso envolve compreender o perfil dos usuários, testar as funcionalidades e adaptar o produto com base nas necessidades reais. Quando uma interface oferece uma boa experiência, os usuários conseguem realizar suas tarefas com mais rapidez e menos frustração, o que contribui para a aceitação e o sucesso do produto (Rogers, Sharp e Preece, 2013).

4.1 DESCRIÇÃO DE CENÁRIO

Conforme Grainer (2018), a construção de cenários narrativos possibilita representar contextos específicos de uso ao considerar fatores como habilidades, limitações, objetivos e ambientes de interação. Essa abordagem ajuda os profissionais de UX a antecipar obstáculos e a desenvolver soluções mais acessíveis e sensíveis à diversidade dos usuários. Os cenários proporcionam uma compreensão completa da experiência do usuário, incorporando motivações, emoções e as dificuldades enfrentadas por pessoas em diferentes contextos. Por meio deles, é possível identificar obstáculos como textos com baixo contraste, ausência de linguagem clara ou falta de suporte.

Esse tipo de narrativa revela pontos críticos que não seriam percebidos apenas por testes funcionais. A partir dessas observações, o design pode ser aprimorado com soluções como modos de alto contraste, botões com ícones intuitivos, legendas e instruções passo a passo. Essas adaptações não surgem por acaso, mas da compreensão empática do contexto real de uso.

Para apresentar um dos cenários de uso do sistema, o Quadro 32 ilustra o processo de contratação de um plano funerário realizado por um cliente em uma funerária parceira. Este cenário foi desenvolvido com o objetivo de simular a interação entre o cliente e a plataforma, destacando as etapas e o fluxo de trabalho necessários para formalizar o contrato e realizar a assinatura digital. Através do quadro, é possível visualizar o contexto no qual o atendente da funerária, devidamente autenticado no sistema, orienta o cliente no preenchimento das informações contratuais, assegurando que o processo ocorra de forma clara, segura e eficiente. A confirmação do sistema e a finalização do contrato representam o encerramento do atendimento, demonstrando a usabilidade da interface e a agilidade proporcionada pela solução.

Quadro 32 – Cenário – Registro de Novo Plano Funerário

No ambiente tranquilo da funerária, um funcionário autenticado no sistema de gerenciamento de planos funerários recebe novos clientes interessados em contratar um plano. Ele registra os dados pessoais, escolhe o tipo de plano e insere as informações contratuais. Após inserir os dados, clica em “Confirmar Plano”, concluindo o processo. Com tudo devidamente registrado, o cliente sai satisfeito e o funcionário retorna à sua rotina, certo de ter proporcionado amparo em um momento importante.

Fonte: Elaborado pelos autores.

O Quadro 33 retrata a interação de um colaborador da empresa funerária com o sistema, tendo como finalidade a análise das informações referentes aos planos ativos e inativos, no contexto de uma solicitação de benefícios vinculada a um cliente falecido. A descrição demonstra de que forma o sistema contribui para o monitoramento eficaz dos planos e para o acompanhamento do desempenho orçamentário, permitindo a emissão de relatórios voltados à gestão financeira.

Quadro 33 – Cenário – Consulta e Validação de Plano Funerário Pós-Óbito

No ambiente reservado da funerária, um colaborador autenticado acessa o sistema de gerenciamento de planos funerários para atender a uma solicitação relacionada a um cliente falecido. Ele verifica o contrato do cliente e verifica o histórico de pagamentos, certificando-se de que todas as obrigações contratuais estejam em conformidade. Após confirmar que os registros financeiros estão em ordem, o funcionário emite um relatório detalhado, que

subsidiará a análise dos benefícios a serem concedidos. A operação é concluída com discrição e eficiência.

Fonte: Elaborado pelos autores.

4.2 DESCRIÇÃO DE PERSONAS

No processo de desenvolvimento de sistemas, o uso de personas é uma prática essencial para garantir que as necessidades e características do público-alvo sejam consideradas. Segundo Unger e Chandler (2009), personas são representações detalhadas de usuários-alvo que auxiliam na compreensão prática de quem utiliza o sistema e de que forma isso ocorre. Criadas a partir de dados reais, elas orientam decisões de design e definição de funcionalidades. Para desenvolver personas eficazes, é fundamental identificar o público e reunir informações suficientes que revelem seus comportamentos, motivações e expectativas.

As personas apresentadas para o projeto LUMENSYS foram criadas com base em informações detalhadas sobre os usuários potenciais. Essas personas incluem dados como foto, nome, idade, localização e ocupação, além de objetivos específicos de uso do sistema (Unger; Chandler, 2009). Tais dados foram utilizados para representar com precisão o perfil dos usuários e para guiar o desenvolvimento do sistema de acordo com as necessidades específicas de cada persona.

A Figura 16 apresenta a persona "Marco Andrade", um gerente operacional de uma funerária de médio porte, responsável pelo monitoramento e pela análise detalhada dos planos funerários, contabilidade e estoque. Com 06 anos de atuação, "Marco" desempenha um papel essencial na gestão financeira da funerária que atua, assegurando a precisão e a integridade dos planos de seus clientes.

Figura 16 – Persona 1



Fonte: Elaborado pelos autores.

A Figura 17 apresenta a persona "Eliane Costa", uma diretora financeira e proprietária de uma funerária tradicional do interior, responsável pelo monitoramento e pela análise detalhada da contabilidade e auditorias. Com 25 anos de atuação, “Eliane” desempenha um papel essencial na gestão financeira da funerária que atua, assegurando a precisão e a integridade da contabilidade de seu estabelecimento.

Figura 17 – Persona 2

| | |
|--|--|
|  <p>Dados da persona</p> <p>Idade: 56</p> <p>Formação acadêmica: Ciências Contábeis</p> <p>Tempo no cargo: 25 anos</p> <p>Localização: Minas Gerais, cidade com 30 mil habitantes</p> <p>Experiência com tecnologia: Básica; utiliza planilhas e sistema legado local</p> | <h2>Eliane Costa</h2> <p>Diretora financeira e proprietária de uma funerária tradicional no interior</p> <p>Objetivos e Motivações</p> <ul style="list-style-type: none"> • Migrar para um sistema intuitivo que simplifique sua rotina • Garantir conformidade fiscal e facilidade nos fechamentos mensais • Reduzir papelada, automatizar cadastros e atualizar contratos sem complicações <p>Frustrações e Desafios</p> <ul style="list-style-type: none"> • Sistema atual desatualizado e difícil de usar • Dificuldade em acompanhar os vencimentos e auditorias • Precisa confiar nos dados financeiros sem depender de terceiros <p>Necessidades do Sistema para Atender Marco</p> <ul style="list-style-type: none"> • Sistema com interface clara e suporte humanizado • Funcionalidade de exportar relatórios contábeis e extratos mensais • Treinamento inicial para equipe • Backup automático e auditoria de acessos |
| <p>Características Comportamentais</p> <ul style="list-style-type: none"> • Perfil comportamental principal: Estável e Cautelosa • Características marcantes: <ul style="list-style-type: none"> • Gosta de rotinas e processos bem definidos • Evita riscos e mudanças abruptas • Leal a fornecedores confiáveis • Prefere segurança e previsibilidade à inovação excessiva • Tem preocupação com clareza e suporte humanizado • Como isso impacta o sistema: <ul style="list-style-type: none"> • Espera que o sistema seja confiável e fácil de aprender • Valoriza assistência próxima na migração e uso inicial • Precisa de apoio para confiar nas funcionalidades digitais • Busca algo que resolva seu problema sem exigir grandes adaptações | |

Fonte: Elaborado pelos autores.

4.3 ESBOÇOS DE TELAS (WIREFRAMES)

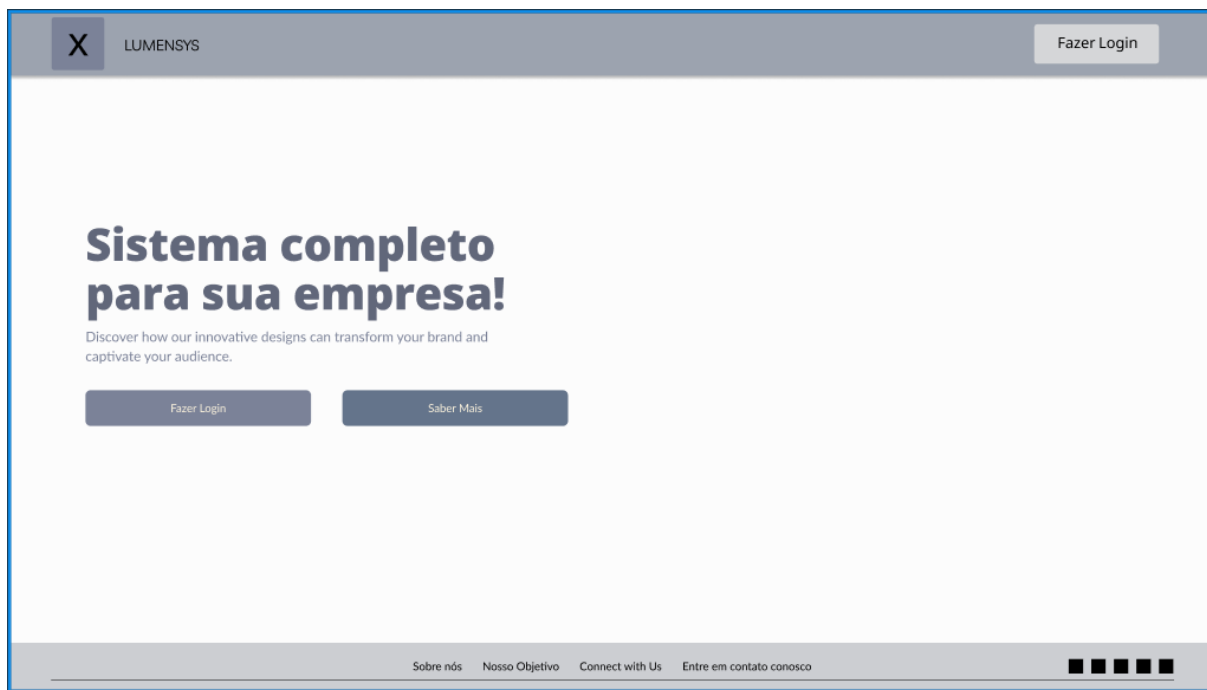
Wireframes são protótipos de baixa fidelidade de páginas web ou telas de aplicações. Eles são utilizados para mapear elementos-chave, como a navegação, seções de conteúdo, uso de imagens e mídias, componentes de formulários e chamadas para ação que serão exibidos na interface final (Unger; Chandler, 2009).

O design de um wireframe é, em geral, feito em escalas de preto, branco e cinza, usando substitutos para imagens e tipografias não especificadas, pois o foco é estrutural e funcional, e não visual. Essa simplicidade é intencional, já que o wireframe serve como uma ferramenta de comunicação para visualizar a disposição dos elementos e obter uma validação inicial de stakeholders antes das etapas de design visual e desenvolvimento (Unger; Chandler, 2009).

A relevância dos wireframes reside em sua capacidade de oferecer uma representação inicial e compreensível do layout do sistema, facilitando o alinhamento das expectativas do cliente e da equipe de desenvolvimento. Assim, a validação dos wireframes pelo cliente permite ajustar a estrutura da interface e funcionalidades propostas com antecedência, evitando retrabalho nas fases posteriores (Unger; Chandler, 2009).

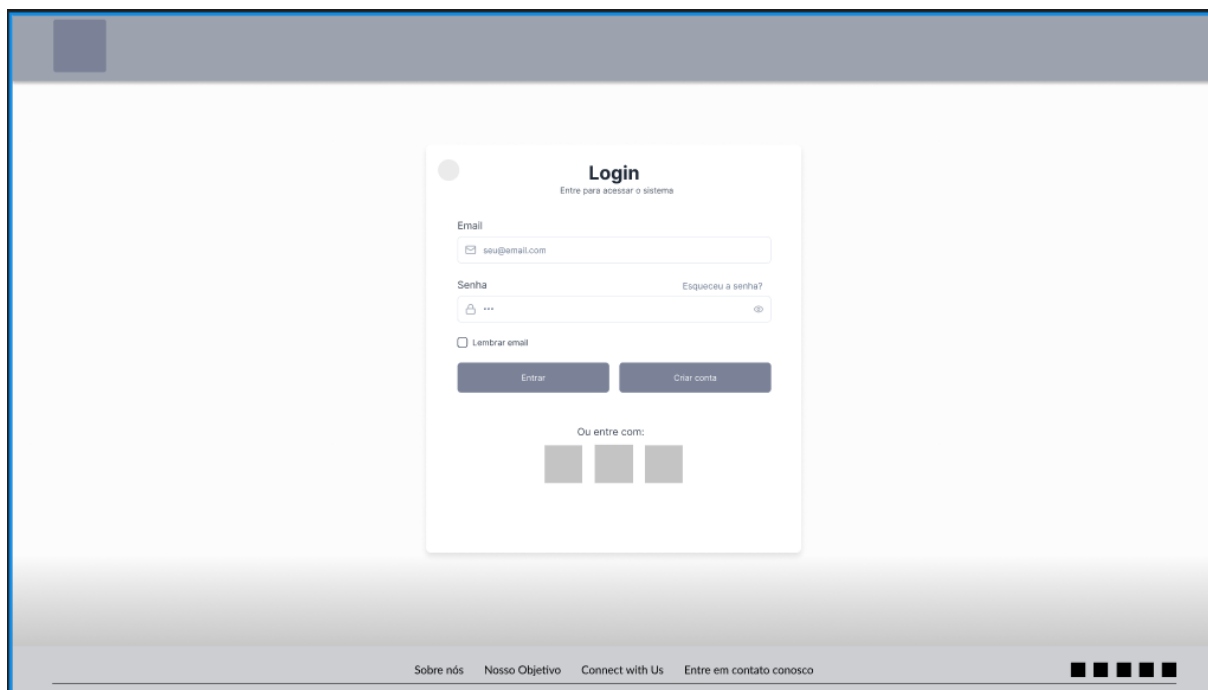
Nas figuras 18 a 23, são apresentados os wireframes das principais telas do sistema. Com base nessas representações, será possível desenvolver o protótipo visual dessas e de outras telas que compõem o sistema LUMENSYS, avançando com segurança na criação da interface final.

Na Figura 18, é apresentado o wireframe da tela inicial. A estrutura exibe um Header na parte superior, que facilita o acesso rápido à funcionalidade de login. Abaixo, alinhado à esquerda, encontra-se um texto de boas-vindas e dois botões: um para login e outro para "Saber Mais", destinado a esclarecer dúvidas sobre as funcionalidades do sistema. Ao final da página, há um Footer, que contém links de navegação e ícones com o objetivo de oferecer informações complementares e facilitar o acesso a seções relevantes do sistema.

Figura 18 – Wireframe – Tela Home

Fonte: Elaborado pelos autores.

Na Figura 19, é apresentado o wireframe da tela de login. O Header segue o mesmo padrão da tela inicial, assegurando uniformidade na experiência de navegação do usuário. O design consistente facilita o aprendizado e a familiarização com o sistema, promovendo uma navegação mais eficiente. No centro da interface, está posicionado um formulário para a realização do login de usuários cadastrados. Esse formulário também apresenta os botões "Entrar" e "Criar Conta", permitindo que o usuário acesse facilmente o sistema. Por fim, na parte inferior da página, encontra-se o Footer, que segue o mesmo padrão adotado em todas as telas (Figura 19).

Figura 19 – Wireframe – Tela de Login

Fonte: Elaborado pelos autores.

Conforme apresentado na Figura 20, o wireframe da tela de cadastro de usuário mantém o padrão de design da tela de login, com o Header garantindo a uniformidade da experiência de navegação do usuário. No centro da interface, o layout segue o modelo da tela de login, apresentando um formulário para acessar uma conta existente e outro para a criação de uma nova conta. Por fim, na parte inferior da página, encontra-se o Footer, que segue o mesmo padrão adotado em todas as páginas.

Figura 20 – Wireframe – Tela de Cadastro de Usuário

The wireframe shows a user registration form titled "Cadastro de usuário" with the subtitle "Entre para acessar o sistema". The form is centered on a light gray background. It includes the following fields and elements:

- Email:** A text input field with a placeholder "seu@email.com" and an email icon on the left.
- Nome de usuário:** A text input field with a placeholder "Nome usuário" and a user icon on the left.
- Senha:** A password input field with a placeholder "..." and a lock icon on the left.
- Confirmar senha:** A password input field with a placeholder "..." and a lock icon on the left.
- Buttons:** Two buttons at the bottom: "Entrar com a conta" (light gray) and "Criar conta" (dark gray).

The footer of the page contains the following links: "Sobre nós", "Nosso Objetivo", "Connect with Us", and "Entre em contato conosco". There are also four black squares on the right side of the footer.

Fonte: Elaborado pelos autores.

O wireframe da tela de cadastro de empresa (Figura 21) segue os mesmos padrões previamente estabelecidos. O Header é mantido, assegurando a uniformidade da experiência de navegação do usuário. No centro da interface, o layout apresenta um formulário maior, contendo um botão para o cadastro de novas empresas. Por fim, na parte inferior da página, encontra-se o Footer, que segue o mesmo padrão adotado em todas as páginas.

Figura 21 – Wireframe – Tela de Cadastro de Empresa

The wireframe shows a company registration form with the following fields and layout:

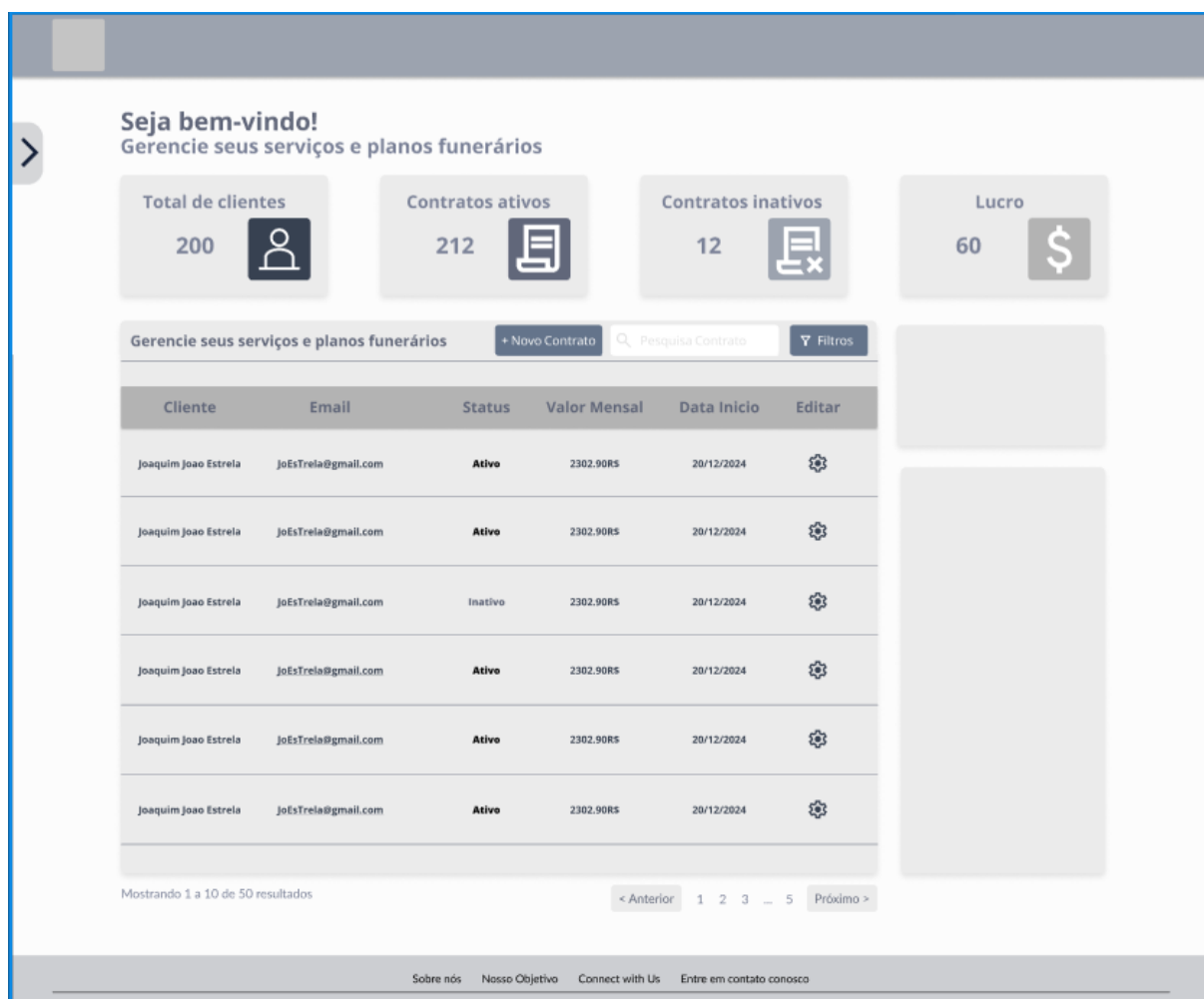
- Header:** A dark gray bar at the top left.
- Title:** "Cadastro de empresa" in a dark gray box.
- Form Fields:**
 - Nome Fantasia: * (text input)
 - Razão Social: (text input)
 - C.N.P.J.: * (text input)
 - Email: * (text input)
 - Telefone: (text input)
 - CEP: (text input)
 - Bairro: (text input)
 - UF: (text input)
 - Endereço: (text input)
- Buttons:** "Voltar" (dark gray) and "Cadastrar" (blue) at the bottom right of the form.
- Footer:** "Sobre nós", "Nosso Objetivo", "Connect with Us", "Entre em contato conosco" and four black squares.

Fonte: Elaborado pelos autores.

O wireframe da tela *Dashboard* (Figura 22) segue os mesmos padrões previamente estabelecidos. O *Header* é mantido para garantir a uniformidade da experiência de navegação do usuário. No topo da tela, há quatro *cards* que informam o estado geral dos contratos. Logo abaixo, encontra-se uma interface que lista os contratos, com três opções: criar um contrato com um cliente, pesquisar contratos pelo nome do cliente e aplicar filtros para refinar a listagem.

À direita da tela, estão posicionados dois campos: um para navegação rápida entre páginas e outro para exibir atividades recentes do sistema, como o cadastro de novos clientes. A interface também inclui uma seção de paginação da lista de contratos, localizada no canto inferior direito, com botões "Anterior" e "Próximo", além da numeração das páginas disponíveis. Por fim, na parte inferior da página, encontra-se o *Footer*, que segue o mesmo padrão utilizado em todas as telas.

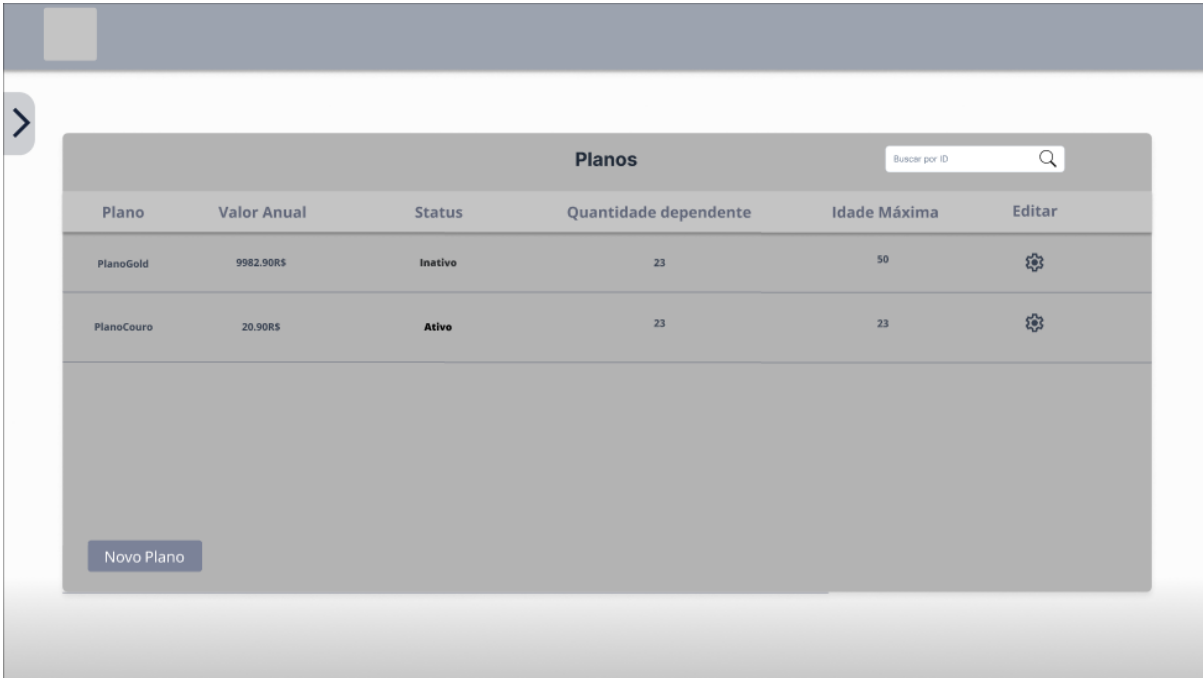
Figura 22 – Wireframe – Tela Dashboard



Fonte: Elaborado pelos autores.

O wireframe da tela de gerenciamento de planos funerários (Figura 23) segue os mesmos padrões previamente estabelecidos. O *Header* é mantido para assegurar a uniformidade da experiência de navegação do usuário. No centro da interface, o layout apresenta uma tabela com informações sobre planos ativos e inativos, além de botões para edição dos planos. Por fim, na parte inferior da página, encontra-se o *Footer*, que segue o mesmo padrão adotado em todas as telas.

Figura 23 – Wireframe – Tela de Gerenciamento de Planos Funerários



| Planos | | | | | |
|------------|-------------|---------|-----------------------|--------------|--------|
| Plano | Valor Anual | Status | Quantidade dependente | Idade Máxima | Editar |
| PlanoGold | 9982.90R\$ | Inativo | 23 | 50 | |
| PlanoCouro | 20.90R\$ | Ativo | 23 | 23 | |

Novo Plano

Fonte: Elaborado pelos autores.

4.4 PROTÓTIPO DE TELA

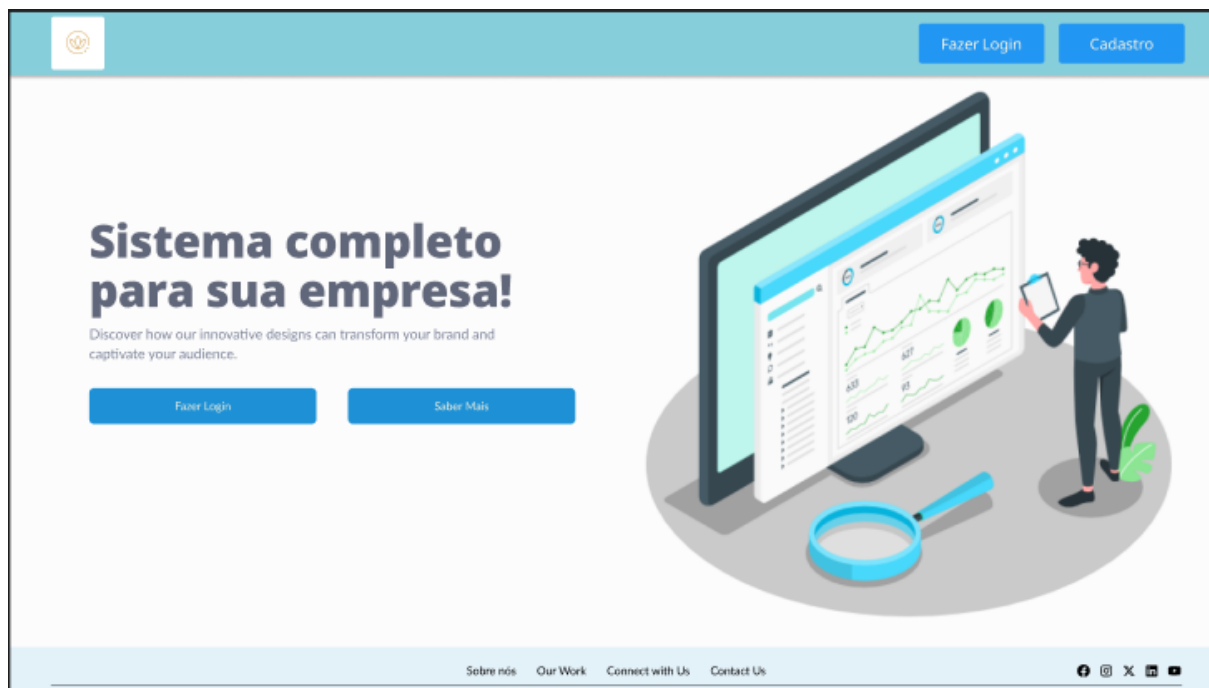
Segundo Rosa (2024), "a finalidade principal de um protótipo é validar hipóteses com uma versão mais simples". Em outras palavras, a criação de protótipos permite testar ideias e verificar, na prática, o que funciona e o que pode ser aprimorado. O wireframe, por sua vez, busca estruturar o layout e a navegação, enquanto o protótipo tem como objetivo simular de forma mais realista a experiência do usuário, possibilitando uma compreensão mais profunda de suas interações com o produto (Rosa, 2024).

Os protótipos podem ser classificados em três tipos principais. O de baixa fidelidade oferece uma versão simplificada e limitada, com foco nos aspectos gerais e no fluxo básico. O de média fidelidade inclui mais detalhes visuais e interativos, embora permaneça incompleto. Já o de alta fidelidade apresenta uma versão avançada, próxima do produto, com elementos gráficos detalhados e funcionalidades interativas, permitindo testar de forma mais realista a experiência e as reações dos usuários (PM3, 2023).

No desenvolvimento dos protótipos de tela do sistema LUMENSYS, foi utilizada a ferramenta Figma (2025). Com ela, foram criados protótipos de alta fidelidade, incorporando cores, elementos gráficos e funcionalidades interativas, a fim de proporcionar uma visualização mais precisa do produto e de validar a experiência do usuário antes da implementação completa.

A interface da (Figura 24) apresenta um design moderno e intuitivo para a página inicial de um sistema empresarial. Com destaque para um slogan central e botões de ação visíveis (Fazer login e Saber Mais, o layout promove fácil navegação e engajamento. Elementos visuais reforçam a proposta tecnológica da plataforma, enquanto o menu inferior oferece acesso rápido a informações institucionais.

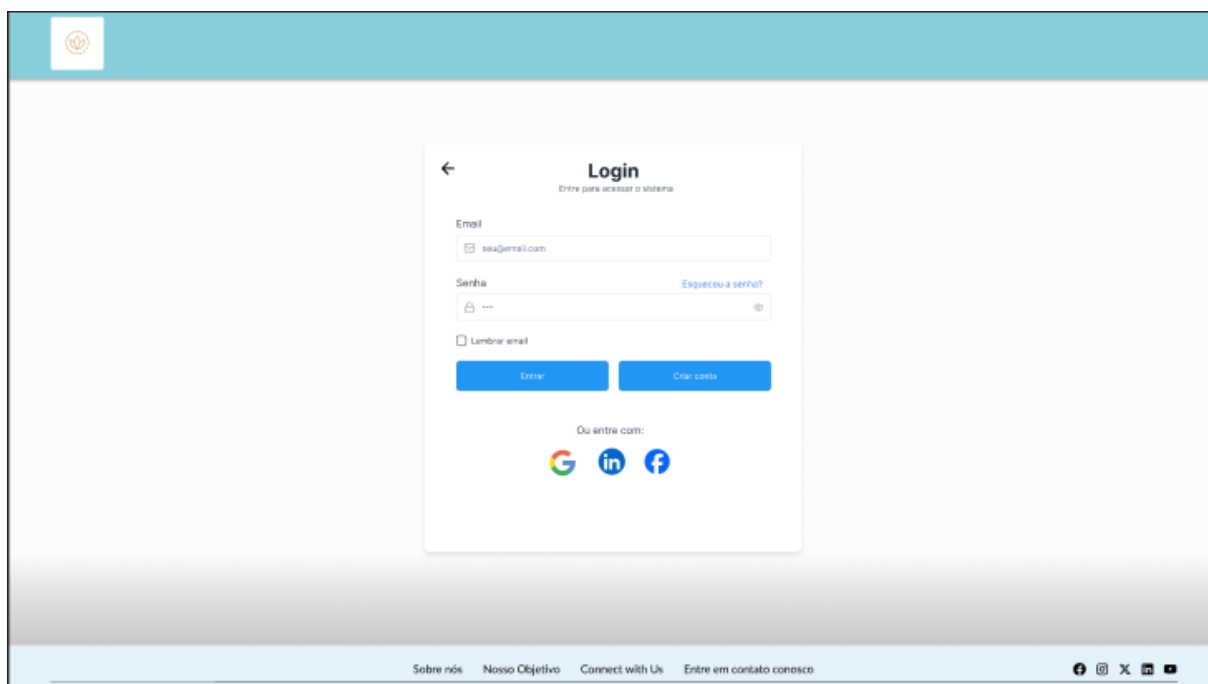
Figura 24 – Protótipo – Tela Home



Fonte: Elaborado pelos autores.

O protótipo da (Figura 25) representa uma interface de login clara e funcional, com campos para inserção de e-mail e senha, opção de lembrar o e-mail, e botões destacados para “Entrar” e “Criar conta”. Também oferece login social via Google, LinkedIn e Facebook. Na parte inferior, há links institucionais que promovem a navegação e o relacionamento com o usuário.

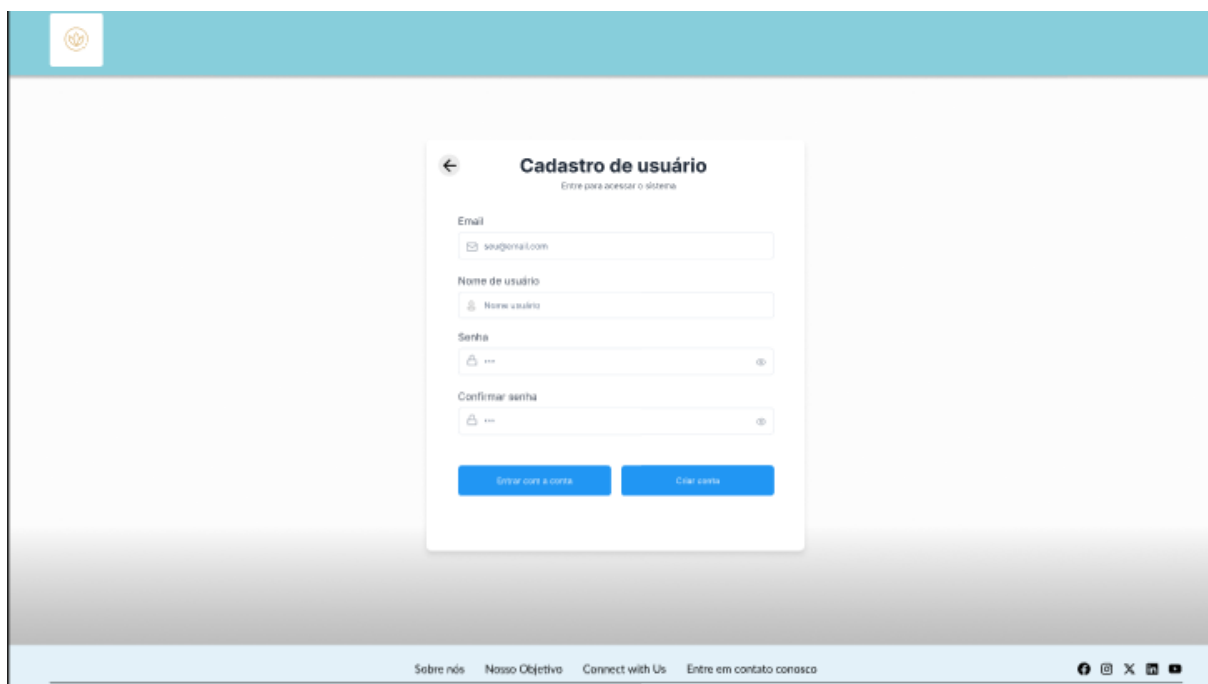
Figura 25 – Protótipo – Tela de Login



Fonte: Elaborado pelos autores.

A interface da (Figura 26) exibe um formulário de cadastro de usuário com campos para e-mail, nome de usuário, senha e confirmação de senha. Os botões “Entrar com a conta” e “Criar conta” oferecem alternativas claras de navegação. A parte inferior inclui links institucionais, promovendo maior acessibilidade e integração com a plataforma.

Figura 26 – Protótipo – Tela de Cadastro de Usuário



Fonte: Elaborado pelos autores.

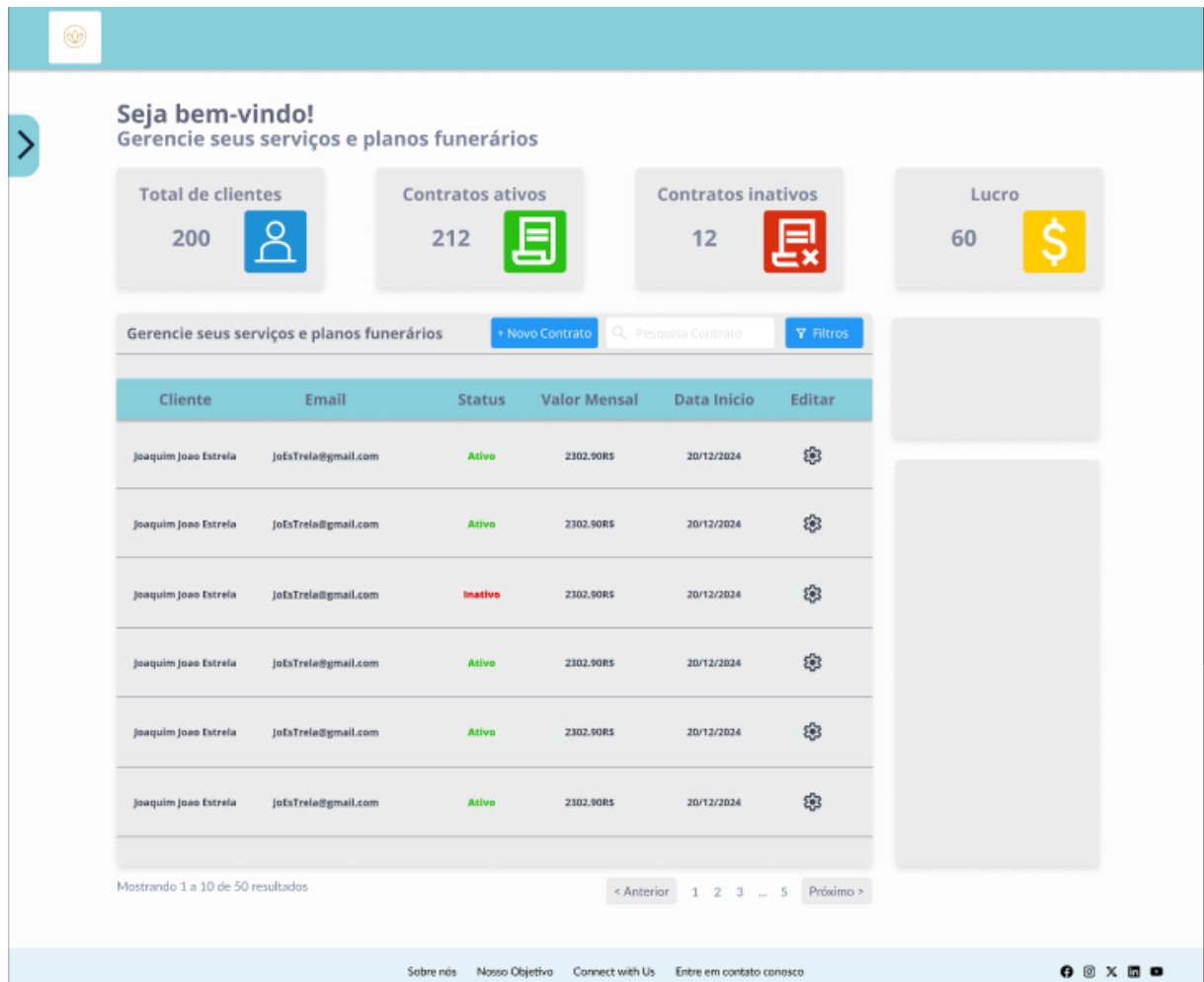
Esta interface da (Figura 27) apresenta um formulário destinado ao cadastro de empresas, com campos essenciais como Nome Fantasia, Razão Social, CNPJ, Email, Telefone, e dados de localização (CEP, Bairro, UF e Endereço). Os botões “Voltar” e “Cadastrar” oferecem uma navegação clara e orientada à ação. O layout é objetivo e funcional, priorizando a coleta eficiente de informações empresariais.

Figura 27 – Protótipo – Tela de Cadastro de Empresa

Fonte: Elaborado pelos autores.

O protótipo da tela dashboard (Figura 28) segue os mesmos padrões previamente estabelecidos. No topo da tela contém quatro cards informando o estado geral da parte dos contratos, logo abaixo conter uma interface listando os contratos, com 3 opções, para criar um contrato com um cliente, para pesquisar por um contrato pelo nome do cliente e uma filtragem de contratos, na direita contém dois campos, um para a navegação rápida para outras páginas e o outro para atividades recentes do sistema, como um novo cliente cadastrado no sistema, há a informação que mostra a paginação da listagem de contratos, na direita, encontram-se os controles de navegação com botões para "Anterior" e "Próximo", além da numeração das páginas disponíveis.

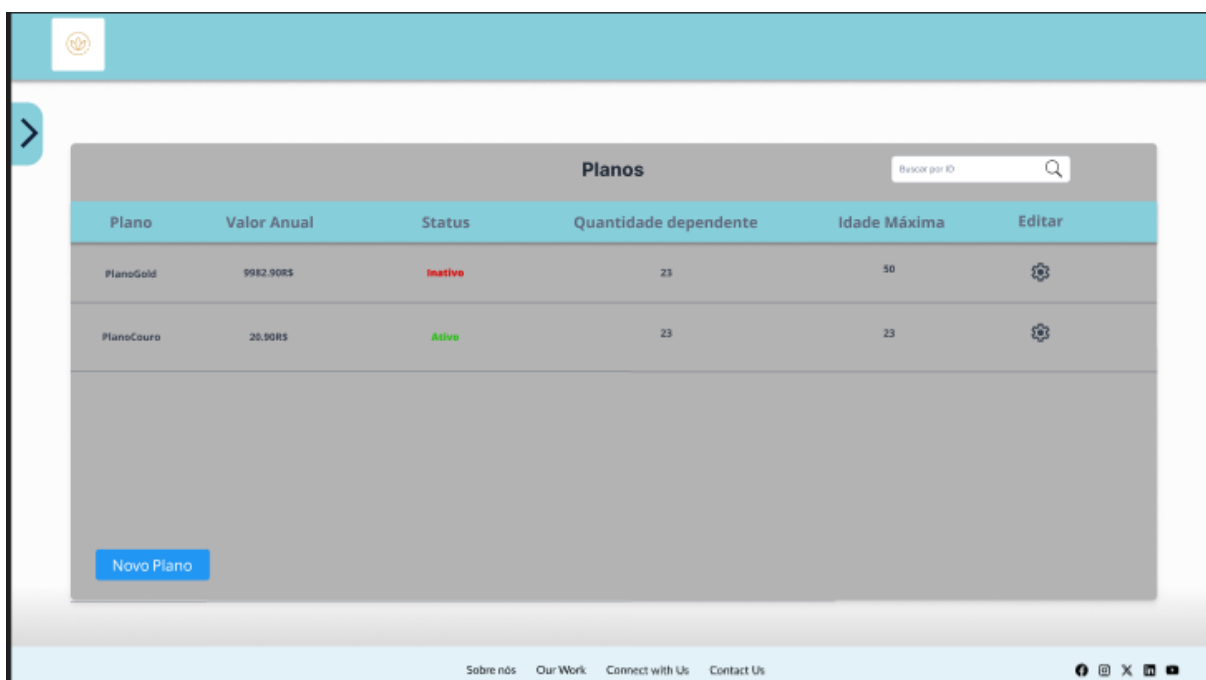
Figura 28 – Protótipo – Tela Dashboard



Fonte: Elaborado pelos autores.

A tela da (Figura 29) exibe uma tabela de planos cadastrados com informações como nome do plano, valor anual, status (ativo ou inativo), número de dependentes permitidos e idade máxima. Dois planos são listados: PlanoGold e PlanoCouro. Há também recursos de busca por ID, botão de edição em cada linha e uma opção destacada para “Novo Plano”, facilitando o controle e a atualização das opções disponíveis na plataforma.

Figura 29 – Protótipo – Tela de Gerenciamento de planos Funerários



Fonte: Elaborado pelos autores.

5 BANCO DE DADOS

O banco de dados é definido por um conjunto de tabelas que possuem dados que representam objetos (Machado, 2020).

O modelo relacional é o principal utilizado para o processamento de dados. Ele organiza as informações em tabelas que representam os dados e suas relações (Silberschatz; Korth; Sudarshan, 2012). Entre os bancos de dados relacionais mais populares, destaca-se o PostgreSQL. Este é um sistema de código aberto, amplamente acessível, do tipo objeto relacional, que oferece diversos recursos para armazenar e processar dados (Silberschatz; Korth; Sudarshan, 2012). No sistema LUMENSYS, adotou-se o banco de dados PostgreSQL para gerenciar e processar o fluxo de dados de forma eficiente. Os bancos de dados realizam diversas operações para acessar e atualizar informações, denominadas transações. Para assegurar que essas transações sejam confiáveis e consistentes, aplica-se o conceito ACID3, que compreende quatro propriedades fundamentais (Silberschatz; Korth; Sudarshan, 2012).

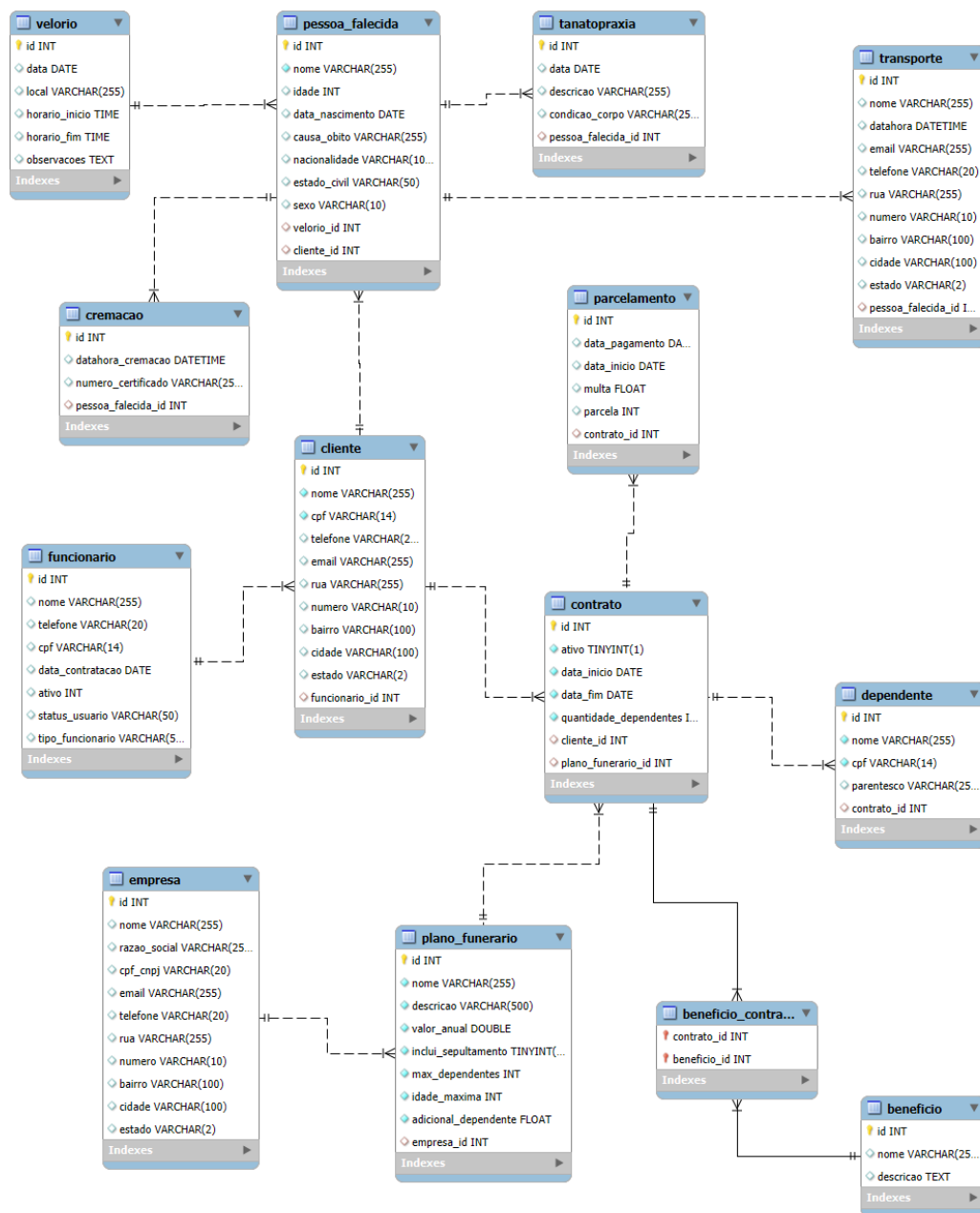
5.1 MODELOS DE ENTIDADE RELACIONAMENTO

O Modelo Entidade-Relacionamento (MER) é uma ferramenta amplamente utilizada para representar a estrutura lógica de um banco de dados de forma clara e simplificada, conforme discutido por Silberschatz, Korth e Sudarshan (2012).

O mapeamento objeto-relacional utiliza elementos como tabelas identificadas por chaves primárias, que servem para individualizar registros, atributos que representam os dados associados a cada registro, e chaves estrangeiras, que estabelecem relacionamentos entre tabelas (Silberschatz; Korth; Sudarshan, 2012).

O mapeamento objeto-relacional das tabelas do sistema LUMENSYS, mostra uma visão sobre os seus atributos e os relacionamentos entre elas. Com essa visualização, a compreensão da estrutura do banco de dados é mais bem identificada, assim destacando as interações entre as tabelas, para a o funcionamento do sistema.

Figura 30 – Modelo Entidade Relacionamento



I

5.2 SCRIPT DE TABELAS

O script das tabelas foi elaborado com base na estrutura apresentada por Machado (2020).

Nos Quadros 34 a 47, são apresentados os scripts do banco de dados do sistema LUMENSYS, que são gerados automaticamente pelo Entity Framework com base na cama de modelo do sistema. Eles definirão os relacionamentos entre os diferentes elementos do sistema e representar cada uma das classes mapeadas na base de dados. Correspondendo cada a criação de uma tabela do sistema LUMENSYS, definindo os atributos e seus relacionamentos. Além disso, seguem as melhores práticas de modelagem relacional, promovendo a integração entre a camada de aplicação e o banco de dados, bem como o desempenho adequado do sistema como um todo. Conforme destaca Machado (2020), a definição adequada da estrutura do banco é essencial para a consistência e a eficiência das operações em sistemas corporativos.

Quadro 34 – Tabela Empresa

```
CREATE TABLE empresa (
    id SERIAL PRIMARY KEY,
    cpf_cnpj VARCHAR(14) NOT NULL,
    nome VARCHAR(100) NOT NULL,
    email VARCHAR(50) NOT NULL,
    telefone VARCHAR(13) NOT NULL,
    rua VARCHAR(100) NOT NULL,
    numero VARCHAR(10) NOT NULL,
    bairro VARCHAR(60) NOT NULL,
    cidade VARCHAR(60) NOT NULL,
    Uf VARCHAR(2) NOT NULL
);
```

Fonte: Elaborado pelos autores.

Quadro 35 – Tabela Plano Funerário

```
CREATE TABLE plano_funerario (
    id SERIAL PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    descricao VARCHAR(500) NOT NULL,
    valor_anual NUMERIC(18,2) NOT NULL,
    disponivel BOOLEAN NOT NULL,
    max_dependentes INTEGER NOT NULL,
    idade_maxima INTEGER NOT NULL,
    adicional_dependente NUMERIC(18,2) NOT NULL,
    empresa_id INTEGER REFERENCES empresa(id)
);
```

Fonte: Elaborado pelos autores.

Quadro 36 – Tabela Usuário

```
CREATE TABLE usuario (
    id SERIAL PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    email VARCHAR(100) NOT NULL,
    senha VARCHAR(100) NOT NULL,
    cpf VARCHAR(11),
    telefone TEXT,
    tipo_usuario INTEGER NOT NULL,
    status_usuario INTEGER NOT NULL,
    data_contratacao DATE,
    empresa_id INTEGER REFERENCES empresa(id)
);
```

```
);
```

Fonte: Elaborado pelos autores.

Quadro 37 – Tabela Cliente

```
CREATE TABLE cliente (  
    id SERIAL PRIMARY KEY,  
    nome TEXT NOT NULL,  
    cpf TEXT NOT NULL,  
    telefone TEXT NOT NULL,  
    email TEXT NOT NULL,  
    rua TEXT NOT NULL,  
    numero TEXT NOT NULL,  
    bairro TEXT NOT NULL,  
    cidade TEXT NOT NULL,  
    uf TEXT NOT NULL,  
    usuario_id INTEGER REFERENCES usuario(id)  
);
```

Fonte: Elaborado pelos autores.

Quadro 38 – Tabela Contrato

```
CREATE TABLE contrato (  
  id SERIAL PRIMARY KEY,  
  ativo BOOLEAN NOT NULL,  
  data_inicio TIMESTAMPTZ NOT NULL,  
  data_fim TIMESTAMPTZ NOT NULL,  
  quantidade_dependentes INTEGER NOT NULL,  
  cliente_id INTEGER REFERENCES cliente(id),  
  plano_funerario_id INTEGER REFERENCES plano_funerario(id)  
);
```

Fonte: Elaborado pelos autores.

Quadro 39 – Tabela Parcelamento

```
CREATE TABLE parcelamento (  
  id SERIAL PRIMARY KEY,  
  data_pagamento DATE,  
  data_inicio DATE NOT NULL,  
  multa NUMERIC(10,2),  
  taxa INTEGER,  
  contrato_id INTEGER NOT NULL REFERENCES contrato(id)  
);
```

Fonte: Elaborado pelos autores.

Quadro 40 – Tabela Pessoa Falecida

```
CREATE TABLE pessoa_falecida (
    id SERIAL PRIMARY KEY,
    nome TEXT NOT NULL,
    idade INTEGER NOT NULL,
    data_nascimento DATE NOT NULL,
    causa_obito TEXT NOT NULL,
    nacionalidade TEXT NOT NULL,
    estado_civil INTEGER NOT NULL,
    sexo INTEGER NOT NULL,
    velorio_id INTEGER REFERENCES velorio(id),
    cliente_id INTEGER REFERENCES cliente(id)
);
```

Fonte: Elaborado pelos autores.

Quadro 41 – Tabela Dependente

```
CREATE TABLE dependente (
    id SERIAL PRIMARY KEY,
    nome TEXT NOT NULL,
    cpf TEXT NOT NULL,
    contrato_id INTEGER REFERENCES contrato(id)
);
```

Fonte: Elaborado pelos autores.

Quadro 42 – Tabela Cremação

```
CREATE TABLE cremacao (
```

```

id SERIAL PRIMARY KEY,
data DATE NOT NULL,
horario TIME NOT NULL,
numero_certificado TEXT NOT NULL,
pessoa_falecida_id INTEGER UNIQUE REFERENCES pessoa_falecida(id)
);

```

Fonte: Elaborado pelos autores.

Quadro 43 – Tabela Tanatopraxia

```

CREATE TABLE tanatopraxia (
id TEXT PRIMARY KEY,
data DATE NOT NULL,
descricao TEXT NOT NULL,
condicao_corpo TEXT NOT NULL,
usuario_id INTEGER REFERENCES usuario(id),
pessoa_falecida_id INTEGER UNIQUE REFERENCES pessoa_falecida(id)
);

```

Fonte: Elaborado pelos autores.

Quadro 44 – Tabela Transporte

```

CREATE TABLE transporte (
id SERIAL PRIMARY KEY,
nome TEXT NOT NULL,
data DATE NOT NULL,
horario TIME NOT NULL,
rua TEXT NOT NULL,
numero TEXT NOT NULL,
bairro TEXT NOT NULL,

```

```

cidade TEXT NOT NULL,
uf TEXT NOT NULL,
pessoa_falecida_id INTEGER REFERENCES pessoa_falecida(id)
);

```

Fonte: Elaborado pelos autores.

Quadro 45 – Tabela Velório

```

CREATE TABLE velorio (
    id SERIAL PRIMARY KEY,
    data DATE NOT NULL,
    local VARCHAR(150) NOT NULL,
    horario_inicio TIME NOT NULL,
    horario_fim TIME NOT NULL,
    observacoes TEXT,
    usuario_id INTEGER REFERENCES usuario(id)
);

```

Fonte: Elaborado pelos autores.

Quadro 46 – Tabela Benefício

```

CREATE TABLE beneficio (
    id SERIAL PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    descricao TEXT NOT NULL
);

```

Fonte: Elaborado pelos autores.

Quadro 47 – Tabela BenefícioContrato

```

CREATE TABLE beneficio_contrato (

```



```
id SERIAL PRIMARY KEY,  
  
contrato_id INTEGER NOT NULL REFERENCES contrato(id),  
  
beneficio_id INTEGER NOT NULL REFERENCES beneficio(id)  
);
```

Fonte: Elaborado pelos autores.

5.3 MAPEAMENTO DE OBJETO RELACIONAL – ORM

Os sistemas de mapeamento objeto-relacional (ORM) baseiam-se em bancos de dados relacionais tradicionais e facilitam a integração entre o modelo de objetos utilizado na programação e o banco de dados relacional na criação de aplicações. Esse tipo de mapeamento permite realizar consultas que são automaticamente convertidas em instruções SQL (Silberschatz, Korth e Sudarshan, 2012).

No projeto LUMENSYS, desenvolvido em C#, foi empregado o Entity Framework como ferramenta para o mapeamento objeto-relacional. Esse recurso exige a instalação de pacotes específicos, a criação de classes de entidade para representar as tabelas e a configuração do banco de dados por meio de comandos.

6 ARQUITETURA DE SOFTWARE

Segundo Sommerville (2018), a arquitetura do software deve ser definida antes da etapa de implementação, pois isso assegura uma conexão clara entre os requisitos do sistema e a estruturação de seus componentes.

Uma arquitetura bem planejada possibilita o desenvolvimento de um sistema eficiente, promovendo uma divisão clara dos componentes, organização e comunicação eficaz entre eles. Além disso, facilita uma implementação direta e objetiva, atendendo às necessidades operacionais e garantindo manutenção simplificada, o que resulta na redução de custos. Esses processos têm como objetivo otimizar o ciclo de vida do sistema e evitar a sobrecarga dos desenvolvedores durante o processo de criação (Martin, 2020).

6.1 ARQUITETURA DE DESENVOLVIMENTO

Para a execução deste projeto, foram realizadas etapas como a coleta de dados, o diagnóstico da situação atual e o levantamento das informações essenciais ao processo de gestão de despesas públicas. Com base na análise dos dados coletados e no estudo das funcionalidades necessárias, procedeu-se ao levantamento dos requisitos para a modelagem e o desenvolvimento do software, seguindo os princípios metodológicos da engenharia de software delineados por Pressman (2011).

Para representar a análise de requisitos, utilizou-se a Linguagem de Modelagem Unificada (UML), uma linguagem de modelagem baseada no paradigma de orientação a objetos (Guedes, 2018). Por meio do software Astah UML (Astah, 2024), foram desenvolvidos diversos diagramas UML que permitiram uma análise detalhada das funcionalidades e dos requisitos operacionais do sistema, facilitando a identificação precisa dos serviços e recursos que a plataforma pode oferecer aos usuários.

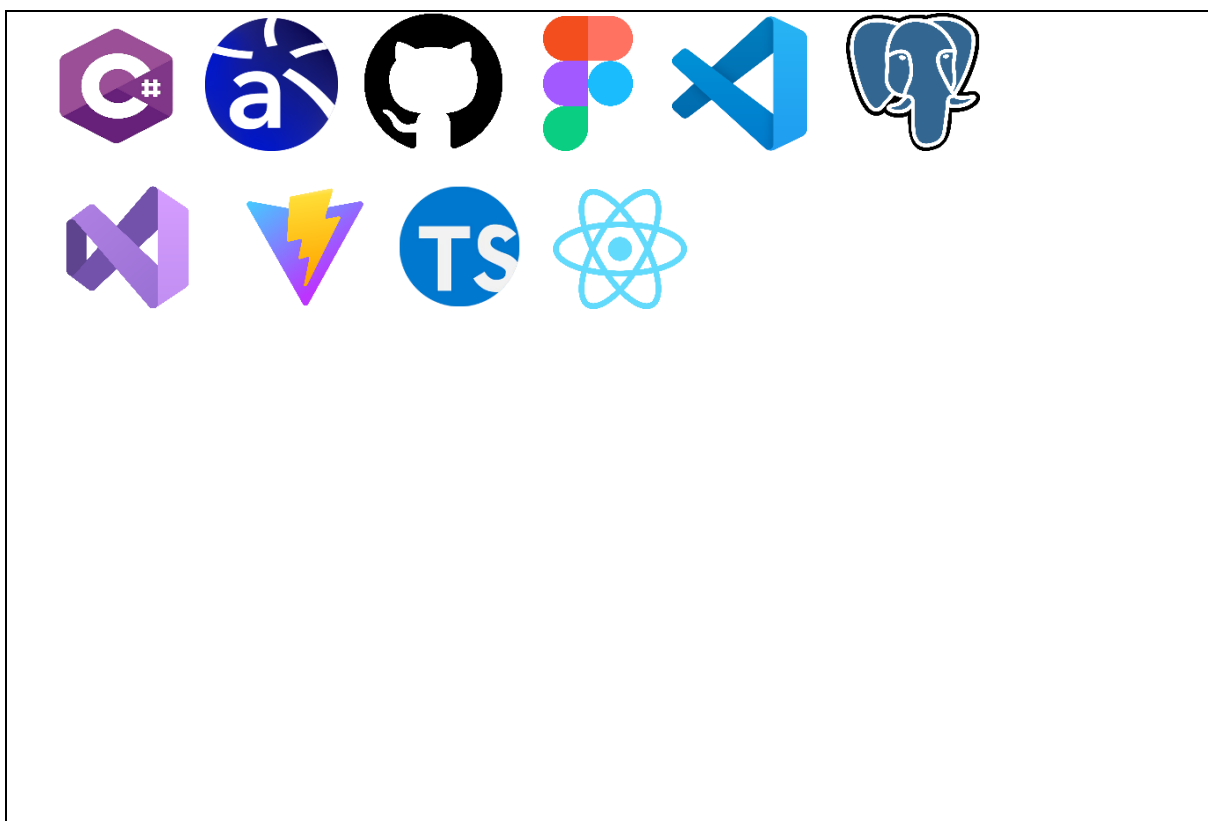
Na fase de desenvolvimento do back-end da aplicação, foi utilizado o Ambiente de Desenvolvimento Integrado (IDE) Visual Studio, reconhecido por suas funcionalidades avançadas de edição de código e recursos robustos para testes. A linguagem de programação selecionada foi o C#, conhecida por ser uma linguagem de uso geral, multiplataforma e de alto desempenho (Wagner, 2024). Para o armazenamento e a gestão dos dados, foi empregado o banco de dados PostgreSQL, um Sistema de Gerenciamento de Banco de Dados Relacional (SGBDR) projetado para administrar o acesso eficiente e seguro às informações (Milani, 2008).

A prototipação das interfaces do sistema foi conduzida por meio da ferramenta Figma (2025). Para o desenvolvimento do front-end, optou-se pela biblioteca React, em conjunto com o empacotador de módulos Vite.js e o framework utilitário de estilos Tailwind CSS, proporcionando uma arquitetura moderna, performática e escalável. Todo o processo de desenvolvimento foi realizado na IDE Visual Studio Code, a qual ofereceu suporte a extensões e funcionalidades que contribuíram para a produtividade da equipe.

A gestão do projeto seguiu os preceitos da metodologia ágil Scrum (SUTHERLAND; SUTHERLAND, 2019), fundamentada na divisão do trabalho em sprints, permitindo entregas incrementais e inspeções contínuas do progresso. O versionamento de código e a centralização das atividades colaborativas foram realizados exclusivamente por meio da plataforma GitHub (2025), que serviu como repositório tanto para o front-end quanto para os demais artefatos documentais vinculados ao projeto.

O Quadro 31 apresenta a representação visual dos softwares gratuitos utilizados ao longo de todo o processo de desenvolvimento do sistema. Estes programas foram empregados nas diferentes etapas do projeto, abrangendo desde a gestão do sistema até a codificação e o design, contribuindo para a organização e eficiência no desenvolvimento do software.

Quadro 48 – Softwares Utilizados



Fonte: Elaborado pelos autores.

6.1.1 BACK-END

Para o desenvolvimento do back-end do projeto LUMENSYS, foi utilizada a linguagem de programação C#, uma linguagem orientada a objetos, multiplataforma e de código aberto (Wagner, 2024). A escolha do C# foi baseada em sua robustez, escalabilidade e flexibilidade, características importantes para o desenvolvimento de sistemas complexos e de fácil manutenção (Wagner, 2024).

A aplicação foi estruturada para fornecer serviços por meio de uma API RESTful, que segue os princípios do Representational State Transfer (REST). Essa arquitetura, amplamente adotada em sistemas modernos, utiliza o protocolo HTTP para realizar operações de CRUD (Create, Read, Update, Delete) sobre recursos, os quais são identificados por URIs (Uniform Resource Identifiers). Cada recurso é tratado como uma entidade única e manipulado por meio dos métodos HTTP padrão, como GET, POST, PUT e DELETE. Essa abordagem permite uma interface padronizada e eficiente, além de garantir a flexibilidade e a escalabilidade necessárias para o gerenciamento de dados do sistema (Sommerville, 2018).

A comunicação entre o cliente e o servidor no sistema é realizada no formato JSON (JavaScript Object Notation), um formato leve e de fácil leitura para a troca de dados. A escolha do JSON foi estratégica, pois facilita a serialização eficiente das informações e assegura a integração entre diferentes sistemas e linguagens de programação. Esse formato também permite que dados sejam compartilhados de forma padronizada e acessível entre plataformas distintas, o que aprimora a interoperabilidade do sistema e torna a comunicação mais fluida, independentemente das tecnologias utilizadas em cada camada da aplicação (Sommerville, 2018).

A arquitetura do sistema foi organizada em camadas para garantir a modularidade e facilitar a manutenção e escalabilidade do código. A camada de Model é responsável por representar os dados e as regras de negócio do sistema, incluindo a estrutura do banco de dados, o que facilita o controle e a integridade dos dados (Baradel, 2024). A camada Repository atua como uma abstração entre a lógica de negócios e o acesso aos dados, separando essas duas responsabilidades e permitindo uma maior flexibilidade na manipulação e persistência das informações (Meine, 2024).

A camada Service, por sua vez, é responsável pela implementação das regras de negócio e pela validação das informações recebidas da camada Controller, garantindo que todos os dados sejam consistentes e atendam aos requisitos da aplicação (Silvestre, 2022).

6.1.2 FRONT-END

Para o desenvolvimento do front-end do LUMENSYS, foi utilizada a biblioteca React em conjunto com o Vite.js, uma ferramenta moderna de build e desenvolvimento frontend que proporciona um ambiente de desenvolvimento leve, rápido e altamente eficiente. A linguagem escolhida foi o TypeScript, um superconjunto estrito de JavaScript, de código aberto e mantido pela Microsoft. O TypeScript adiciona tipagem estática opcional, permitindo maior previsibilidade na manipulação de dados e facilitando a detecção de erros em tempo de compilação, o que contribui para a padronização, reutilização e organização dos componentes da aplicação.

O React é uma biblioteca JavaScript de código aberto criada e mantida pelo Facebook, projetada para a construção de interfaces de usuário com foco em componentização, performance e reatividade. Uma das principais características do React é a sua arquitetura baseada em componentes, que permite dividir a interface em pequenas partes independentes e reutilizáveis. Essa abordagem favorece a organização do código, facilita testes unitários e promove a manutenção e escalabilidade do sistema. Cada componente pode ser sem estado ou com estado, possibilitando a criação de interfaces dinâmicas que reagem a mudanças nos dados em tempo real, sem necessidade de recarregar a página.

A estrutura do front-end foi organizada com base em componentes reutilizáveis, favorecendo a modularização do código e permitindo que diferentes partes da interface fossem compartilhadas entre várias páginas do sistema. Para o controle de navegação, foi utilizada a biblioteca React Router DOM, que oferece suporte a rotas dinâmicas e navegação entre páginas sem recarregamento da interface, promovendo uma experiência fluida e contínua para o usuário.

A estilização da interface foi realizada com o uso do Tailwind CSS, uma biblioteca utilitária de classes que permite escrever estilos diretamente no HTML/JSX, reduzindo a necessidade de folhas de estilo separadas. Essa abordagem torna o desenvolvimento visual mais rápido e consistente, ao mesmo tempo que facilita a criação de interfaces responsivas e modernas.

Para a comunicação com APIs e serviços externos, foi adotada a biblioteca Axios, que fornece uma API baseada em Promises para realizar requisições HTTP de maneira simplificada. Por meio do Axios, foram implementadas operações como GET, POST, PUT e DELETE, assegurando a integração eficiente entre o front-end e o back-end da aplicação.

Além disso, a biblioteca js-cookie foi utilizada para o gerenciamento de cookies no navegador, permitindo o armazenamento e a recuperação de informações como tokens de autenticação e preferências do usuário. Isso foi essencial para a manutenção de sessões de usuário e controle de acesso a funcionalidades protegidas.

A integração entre essas tecnologias React, Vite.js, TypeScript, Tailwind CSS, Axios, React Router DOM e js-cookie resulta em um front-end moderno, modular, responsivo e seguro, capaz de oferecer uma experiência de usuário otimizada e uma base sólida para futuras evoluções da aplicação LUMENSYS.

6.2 SEGURANÇA DA INFORMAÇÃO

A segurança da informação constitui um domínio fundamental no campo da tecnologia da informação, cujo objetivo principal é assegurar a proteção de dados e informações estratégicas, de modo a garantir que apenas usuários devidamente autorizados tenham acesso a tais conteúdos (Holdsworth; Kosinski, 2024). Segundo os autores supracitados, a segurança da informação (InfoSec) compreende a salvaguarda das informações, preservando sua confidencialidade, integridade e disponibilidade, em consonância com as demandas dos usuários legitimamente autorizados. Tal abordagem visa prevenir o acesso e a modificação indevida de dados, assegurando simultaneamente sua disponibilidade aos usuários habilitados sempre que necessário.

Os fundamentos do InfoSec estão alicerçados na denominada tríade CIA *Confidentiality, Integrity, and Availability*. A Confidencialidade refere-se à restrição de acesso a informações sensíveis a indivíduos não autorizados. A Integridade diz respeito à preservação da precisão e completude dos dados, impedindo alterações não autorizadas. A Disponibilidade, por sua vez, assegura que os dados estejam acessíveis aos usuários autorizados, no momento em que forem requeridos (Holdsworth; Kosinski, 2024).

Adicionalmente, Holdsworth e Kosinski (2024) enfatizam dois princípios complementares: a Garantia da Informação, que envolve a implementação contínua dos elementos da tríade CIA, e a Não Repudição, cuja finalidade é evitar que um usuário negue a realização de determinada ação dentro do sistema, uma vez que todas as interações são devidamente autenticadas e registradas.

No escopo das soluções voltadas à proteção de dados, destaca-se o uso do JSON Web Token (JWT, 2024), tecnologia amplamente empregada para assegurar a comunicação segura entre diferentes componentes de um sistema. O JWT permite a transmissão de informações por meio de objetos no formato JSON, sendo comumente utilizado em contextos de autenticação e autorização. Após o processo de autenticação inicial, o servidor gera e retorna um token ao cliente; em requisições subsequentes, este token é incluído no cabeçalho das solicitações, possibilitando a verificação da identidade do usuário e o controle de acesso aos recursos previamente autorizados (JWT, 2024).

No sistema LUMENSYS, o processo de autenticação ocorre por meio do fornecimento de credenciais, como e-mail e senha. Após a validação dessas informações, um JWT é gerado e entregue ao usuário, funcionando como uma chave de acesso às funcionalidades permitidas. Tal abordagem possibilita a comunicação segura e eficiente entre cliente e servidor, sem a necessidade de repetir o processo de autenticação a cada nova solicitação.

No âmbito das medidas de segurança, o LUMENSYS adota mecanismos de auditoria como recurso complementar. A auditoria consiste no registro sistemático e detalhado das atividades realizadas no sistema, incluindo acessos e alterações, permitindo a elaboração de relatórios consistentes sobre a movimentação interna (Pressman; Maxim, 2021). Esse processo contempla a documentação de dados como valores anteriores e modificados, datas, horários e identificação do responsável por cada ação realizada.

A adoção de tais práticas promove a transparência das operações, facilita a identificação de eventuais irregularidades e assegura a responsabilização pelos atos executados. Além disso, os registros oriundos do processo de auditoria podem ser explorados para análises aprofundadas, tais como a identificação de padrões comportamentais e a detecção de atividades anômalas.

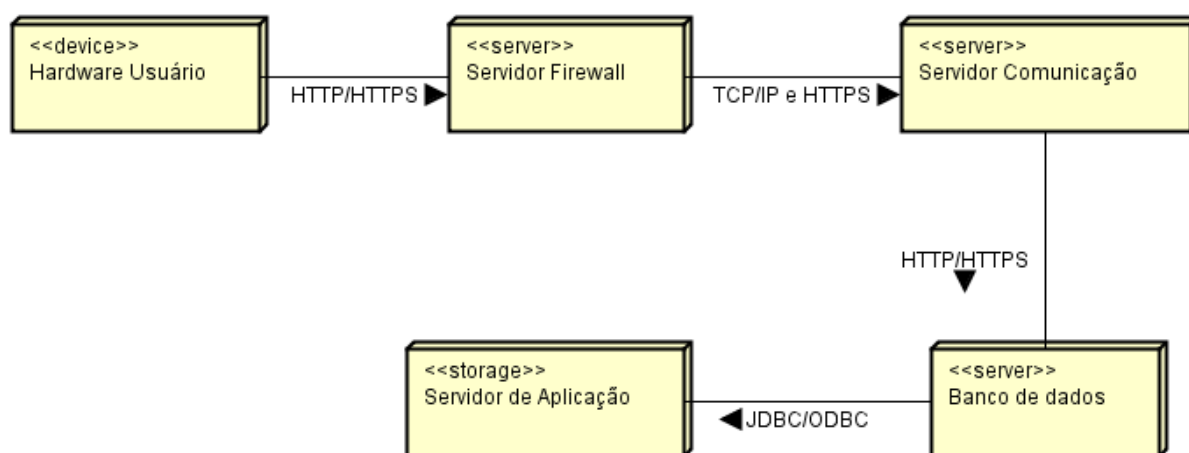
6.3 IMPLANTAÇÃO DO SISTEMA

O diagrama de implantação tem como principal objetivo representar a disposição física dos componentes de software em uma infraestrutura computacional, evidenciando a arquitetura de execução da aplicação. Segundo Guedes (2018), o diagrama de implantação é utilizado para descrever a distribuição dos elementos de software sobre os nós de hardware, incluindo servidores, dispositivos de rede, contêineres e conexões de comunicação.

Na aplicação em questão, o diagrama de implantação descreve a estrutura lógica de hospedagem da solução desenvolvida em dotnet 8, composta por três principais camadas: cliente, backend e banco de dados, todas orquestradas por meio do Docker. A camada cliente é representada pelos dispositivos de acesso dos usuários finais, como computadores pessoais, tablets ou smartphones, que se conectam à aplicação por meio de navegadores web, utilizando os protocolos HTTP ou HTTPS.

Essa comunicação inicial é direcionada ao servidor de firewall, responsável por filtrar e proteger o tráfego de entrada e saída da rede, garantindo que apenas conexões autorizadas possam acessar os serviços internos. Após a validação das requisições, estas são encaminhadas ao servidor de comunicação, que funciona como intermediador entre o cliente e os demais componentes da aplicação. Esse servidor realiza o roteamento das chamadas, válidas sessões e gerencia a comunicação com os contêineres responsáveis pelo processamento da lógica de negócio.

Figura 31 – Diagrama de Implantação



Fonte: Elaborado pelos autores.

O hardware do usuário é classificado como um dispositivo de acesso. Compreende equipamentos como computadores, smartphones ou tablets, que possibilitam a interação com o sistema, normalmente por meio de navegadores web ou aplicativos dedicados. A comunicação entre esses dispositivos e o ambiente da aplicação ocorre por meio dos protocolos HTTP ou HTTPS, o que assegura a troca de informações de forma segura, com tráfego de dados criptografado sempre que necessário.

O servidor de firewall, por sua vez, é responsável por proteger a rede interna contra acessos não autorizados, atuando como uma barreira de segurança entre o ambiente externo e os componentes internos do sistema. Esse servidor monitora, filtra e controla o tráfego de entrada e saída, permitindo apenas conexões seguras e autorizadas. Após a validação das requisições, estas são encaminhadas ao servidor de comunicação por meio dos protocolos TCP/IP e HTTPS, promovendo a integridade do fluxo de dados.

O servidor de comunicação funciona como uma camada intermediária entre a interface de apresentação e os serviços internos da aplicação. Ele é encarregado de gerenciar as requisições recebidas, realizar autenticações de usuários e encaminhar as chamadas para os serviços apropriados. Sua comunicação com os demais componentes ocorre utilizando o protocolo HTTPS, o que contribui para a proteção e a eficiência no tráfego de informações dentro do sistema.

O servidor de aplicação constitui o núcleo central onde reside a lógica de funcionamento do sistema. Nele são processadas as solicitações dos usuários, executadas as regras de negócio e gerenciado o fluxo de informações entre a interface do usuário e o banco de dados. No diagrama, nota-se que esse servidor se comunica com o servidor de comunicação por meio de HTTP/HTTPS, e com o banco de dados por meio do protocolo JDBC/ODBC, possibilitando o acesso estruturado e seguro às informações necessárias para a operação do sistema.

O banco de dados é o componente responsável pelo armazenamento persistente das informações manipuladas pelo sistema. Esse servidor pode ser implementado com sistemas gerenciadores como SQL Server, Oracle ou PostgreSQL, e é acessado tanto pelo servidor de aplicação quanto, em determinados contextos, pelo servidor de comunicação, utilizando protocolos seguros como HTTP/HTTPS.

No backend, foi utilizada uma imagem oficial do SDK do .NET 8 para realizar o processo de build e publicação da aplicação. Posteriormente, os arquivos compilados foram transferidos para uma imagem baseada no runtime do dotnet 8, garantindo a execução eficiente e leve da aplicação. A configuração da string de conexão com o banco de dados foi feita por meio de variáveis de ambiente, permitindo maior flexibilidade e segurança na comunicação com o serviço de dados.

O frontend, desenvolvido com ferramentas baseadas em JavaScript moderno, foi empacotado a partir de uma Figura Node.js, com posterior publicação para uma imagem Nginx. A comunicação com a API foi configurada dinamicamente por meio de variáveis de ambiente, facilitando a adaptação entre ambientes de desenvolvimento e produção. O Nginx também foi configurado para redirecionar requisições da interface para a API de forma transparente, mantendo uma única origem de acesso.

O banco de dados utilizado foi o PostgreSQL, também executado em contêiner, com persistência de dados configurada por volumes. O banco foi iniciado com um script de criação prévio, permitindo a automação da carga inicial de dados e a integridade do ambiente.

A orquestração dos serviços foi feita com o Docker Compose, definindo a ordem de inicialização e dependência entre os contêineres. Com isso, foi possível garantir que a aplicação estivesse disponível apenas após o banco de dados estar devidamente pronto para operação. Essa abordagem proporciona maior controle sobre o ambiente, além de facilitar a manutenção e replicação do sistema em outros servidores ou ambientes de homologação.

7 CONCLUSÃO

A proposta deste trabalho consistiu no desenvolvimento de um software para apoiar a gestão de planos funerários, oferecendo uma solução eficiente e acessível para o controle de contratos, pagamentos, beneficiários e serviços prestados. Durante o processo de desenvolvimento, foi necessário realizar uma análise minuciosa para identificar as informações mais relevantes para exibição e cadastro, com o objetivo de criar uma ferramenta que atendesse às necessidades operacionais e gerenciais do setor funerário.

Além de permitir o cadastro detalhado de planos, clientes e dependentes, o sistema possibilita o acompanhamento do status dos pagamentos, da vigência dos contratos e da utilização dos serviços, tudo de forma ágil e intuitiva. Também foram desenvolvidos dashboards com dados relevantes, como indicadores de inadimplência, categorias de planos mais contratadas e a distribuição regional dos clientes. Esses recursos facilitam a análise estratégica, permitindo uma visão ampla e organizada da operação do negócio.

Para futuras evoluções, foram identificadas melhorias que podem agregar ainda mais valor ao sistema. Entre elas, destaca-se a criação de uma interface para geração de relatórios com periodicidades variadas semanal, bimestral, trimestral, semestral e anual, assim proporcionando uma visão detalhada sobre receitas, perdas, cancelamentos e projeções de crescimento. Essa funcionalidade apoiaria a tomada de decisões gerenciais, otimizando a alocação de recursos e orientando ações voltadas à sustentabilidade do serviço.

No estágio atual, o sistema atende às principais demandas operacionais para o gerenciamento de planos funerários, oferecendo funcionalidades que facilitam o controle de clientes, planos e serviços. Um dos diferenciais mais relevantes da solução desenvolvida é a possibilidade de utilização por duas ou mais empresas pertencentes ao mesmo proprietário, mesmo que estejam localizadas em estados diferentes. Isso amplia significativamente o alcance da ferramenta, permitindo a centralização da gestão administrativa em um único sistema, mesmo em contextos geográficos distintos.

Com as melhorias planejadas, o projeto poderá evoluir incorporando novas funcionalidades, como a ampliação de perfis de acesso, incluindo Funcionário e Visitante e a flexibilização de permissões específicas para cada unidade empresarial cadastrada. Essas evoluções promoverão uma gestão ainda mais integrada, transparente e eficiente, contribuindo para a padronização de processos e o fortalecimento estratégico das empresas do grupo, resultando em melhores serviços aos associados e maior controle por parte dos gestores.

8 REFERÊNCIAS

AKEMI, A. **HTTP: um guia completo sobre o que é e como funciona o protocolo da web.** 2024. Disponível em: <https://www.alura.com.br/artigos/http>. Acesso em 26 mar. 2025.

ANICHE, M. **Orientação a Objetos e SOLID para Ninjas: Projetando Classes Flexíveis.** São Paulo: Casa do Código, 2011.

ASTAH. The Best UML Diagramming Tool Available. Disponível em: <https://astah.net/products/astah-uml/>. Acesso em: 28 fev. 2025.

AXIOS. Axios – Documentação. Disponível em: <https://axios-http.com/ptbr/docs/intro>. Acesso em: 16 jun. 2025.

BARADEL, L. **MVC – O que significa essa sigla e suas camadas.** Disponível em: <https://lucasbaradel.medium.com/mvc-o-que-significa-essa-sigla-e-suas-camadas-142615b78c81>. Acesso em 24 jan. 2025.

CASTRO, E.P.D.; MURILLO, B.K.B. **Implementación de un Sistema Web Para la Gestión de Procesos Operativos de la Funeraria Cosmopolita en el Cantón Milagro.** 2021. Trabalho de graduação (Engenheiro em Computação e Informática) – Universidad Agraria del Ecuador Facultad de Ciencias Agrarias, Milagro, Ecuador 2021.

CARDOSO, A. P. M; SANTOS, T. A. **O Atendimento Humanizado Como Planejamento Estratégico Do Serviço Funerário.** 2020. Pós- Graduação Latu Sensu-MBA em Gestão de Pessoas – UNICEUNA – Centro Universitário Natalense, Natal, 2020.

DSPACE EN ESPOL. **Siestema De Control Administrativo Para La Funeraria “León”: Manual De Diseño.** Disponível em: <http://www.dspace.espol.edu.ec/handle/123456789/58658>. Acesso em 27 mar. 2025.

G1 Globo. Disponível em: <https://g1.globo.com/rs/rio-grande-do-sul/1-ano-de-enchente-rs/noticia/2025/04/29/antes-e-depois-um-ano-apos-enchente-no-rs-veja-como-estao-lugares-atingidos-pela-inundacao.ghtml> . Acesso em 02 jun. 2025

G1 Globo. Disponível em: <https://g1.globo.com/sp/presidente-prudente-regiao/noticia/2020/06/19/coronavirus-modifica-servicos-funerarios-e-impacta-a-forma-de-se-despedir-de-um-ente-querido-e-uma-coisa-muito-fria.ghtml> Acesso em 05 jun. 2025.

GAZETA DO POVO. Disponível em: <https://www.gazetadopovo.com.br/mundo/israel-realiza-funerais-para-enterrar-soldados-mortos-na-guerra-contra-o-hamas/>. Acesso em 02 jun. 2025.

HEBER, L.M.V.; DAVID, G.H. **Desarrollo y Aplicación de un Sistema Orientado a la Web Para la Sistematización de la Previccion Exequial de los Procesos Manuales que Lleva la Funeraria el Rosal en Planeta Rica Córdoba**. 2015. Trabalho de graduação (Engenheiro de Sistemas) – Universidad De Córdoba Facultad De Ingeniería De Sistemas Y Telecomunicaciones, Planeta Rica, Colômbia 2015.

LIVROS DE MARKETING. **Resumo e dicas dos melhores livros de marketing**. Disponível em: <https://www.livrosdemarketing.com.br/marketing/resumo-livro-administracao-de-marketing/>. Acesso em 22 mai. 2025.

MS SISTEMAS. Simplifique sua gestão com a MS SISTEMAS. Disponível em: <https://mssistemas.com.br/site/>. Acesso em 05 jun. 2025.

PROGEM. Software para a gestão de funerárias e planos assistenciais. Disponível em: <https://progem.com.br/>. Aceso em 22 de mai. 2025.

[SCRIBD]. Lar dos documentos do mundo. Disponível em: <https://pt.scribd.com/document/821341122/Sistemas-de-Informacao-Gerencial-Laudon-Resumo>. Acesso em 22 mai. 2025.

TOTVS. O melhor sistemas de gestão de funerária. Disponível em: <https://www.totvs.com/gestao-deathcare/> . Acesso em 22 mai. 2025.

SOMMERVILLE, I. **Engenharia de software**. 10. ed. São Paulo: Pearson Education do Brasil, 2018.

NIELSEN, Jakob. (1993) **Usability Engineering**. San Francisco: MORGAN Kaufmann Publisher, Inc., 1993.

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

PRESSMAN, R. S. **Engenharia de software**. 7.ed. Porto Alegre: AMGH, 2011.

FIGMA. Pense grande. Crie mais rápido. Disponível em: <https://www.figma.com/>. Acesso em: 17 mar. 2025.

GITHUB. Build and ship software on a single, collaborative platform. Disponível em: <https://github.com>. Acesso em: 15 fev. 2025.

GRAINER, Ste. Inclusive Design – UXcellence. Medium, 3 abr. 2018. Disponível em: <https://medium.com/uxcellence/inclusive-ux/home>. Acesso em: 15 jun. 2025.

GUEDES, R. UML 2: Uma abordagem prática. 3. ed. 2018.

HOLDSWORTH, J; KOSINSKI, M. O que é segurança da informação? Disponível em: <https://www.ibm.com/br-pt/topics/information-security#:~:text=Seguran%C3%A7a%20da%20informa%C3%A7%C3%A3o%20%28InfoSec%29%20%C3%A9%20a%20prote%C3%A7%C3%A3o%20de,usu%C3%A1rios%20autorizados%2C%20permane%C3%A7am%20confidenciais%20e%20mantenham%20sua%20integridade>. 24 jul. 2024.

JS-COOKIE. js-cookie – npm. Disponível em: <https://www.npmjs.com/package/js-cookie>. Acesso em: 16 jun. 2025.

JWT. Introduction to JSON Web Tockes. Disponível em: <https://jwt.io/introduction>. Acesso em: 8 nov. 2024.

MARTIN, R. C. Arquitetura Limpa. O Guia do Artesão para Estrutura e Design de Software. Tradução: Samantha Batista. Rio de Janeiro: Alta Books, 2020. Tradução de: Clean Architecture: A Craftsman's Guide to Software Structure and Design.

MACHADO, F. N. R. **Banco de Dados: Projeto e Implementação**. 4. ed. São Paulo: Érica, 2020.

MICHELETTO, C. Vite.js - O build tool que vai facilitar a sua vida. 2020. Disponível em: <https://dev.to/lixetto/vite-js-o-build-tool-que-vai-facilitar-a-sua-vida-15ho>. Acesso em: 15 jun. 2025.

MICROSOFT. Entity Framework. Disponível em: <https://learn.microsoft.com/pt-br/ef/>. 13 jul. 2025.

MILANI, A. **PostgreSQL: guia do programador**. São Paulo: Novatec Editora, 2008.

PM3. **Prototipação: o que é, tipos, benefícios e etapas do processo**. Disponível em: <https://www.cursospm3.com.br/blog/prototipacao-o-que-e/>. 14 out. 2023.

PRESSMAN, R. S. **Engenharia de software: uma abordagem profissional**. 7. ed. Porto Alegre: AMGH, 2011.

PRESSMAN, R. S; MAXIM, B. R. **Engenharia de Software: uma abordagem profissional**. 9. ed. Porto Alegre: AMGH, 2021.

REACT. React – Documentação Oficial. Disponível em: <https://react.dev/>. Acesso em: 16 jun. 2025.

REACT ROUTER. React Router. Disponível em: <https://reactrouter.com/>. Acesso em: 16 jun. 2025.

ROGERS, Y; SHARP, H; PREECE, J. **Design de Interação – Além da interação humano computador**. 3. ed. Porto Alegre: Bookman, 2013.

ROSA, A. Protótipo: entenda o que é, tipos, exemplos e como fazer na prática. Disponível em: <https://softdesign.com.br/blog/prototipo-baixa-e-alta-fidelidade/#Qual-a-diferenca-entre-um-wireframe-e-prototipo>. 25 set. 2025.

SANTANA, B. **O Que é o Protocolo TCP/IP e Como Ele Funciona?** Disponível em: <https://www.hostinger.com.br/tutoriais/tcp-ip>. 01 ago. 2025.

SILBERSCHATZ, A; KORTH, H. F; SUDARSHAN, S. **Sistema de Banco de Dados**. 6. ed. Rio de Janeiro: Elsevier, 2012.

SILVESTRE, G. **Controller e Service - Uma breve introdução**. Disponível em: <https://dev.to/gabrielhsilvestre/controller-e-service-uma-breve-introducao-24hk>. 25 mar. 2025.

SUTHERLAND, J.; SUTHERLAND, J. J. **Scrum: A arte de fazer o dobro do trabalho na metade do tempo**. 1. ed. São Paulo: Sextante, 2019.

TAILWINDCSS. Tailwind CSS. Disponível em: <https://tailwindcss.com/>. Acesso em: 16 jun. 2025.

UNGER, R; CHANDLER, C. **O Guia para projetar UX**. Rio de Janeiro: Alta Books, 2009. VITEJS. Getting Started | Vite - Disponível em: <https://vite.dev/guide/> . Acesso em 12 jun. 2025.

UNYMOS - ERP e Consultoria para Funerárias e Cemitérios | Gestão Inteligente - Sistema Completo para Funerárias e Cemitérios. Disponível em: <https://unymos.com/> . Acesso em 25 mar 2025

WAGNER, B. **Um tour pela linguagem C#**. Disponível em: <https://learn.microsoft.com/pt-br/dotnet/csharp/tour-of-csharp/>. 09 mai. 2025.

WALKER, J. Desvendando e entendendo as diferenças entre: DTO's, interfaces e types – Um guia para escolher a abordagem certa. Disponível em:

<https://medium.com/@jhonywalkeer/desvendando-e-entendendo-as-diferen%C3%A7as-entre-dtos-interfaces-e-types-um-guia-para-escolher-a-ddb7cd84eb47>. 01 ago. 2023.