

**Lemuel Prates Sobrinho Ferreira**

**Maria Clara Borges Cardoso**

**Vinicius Lúcio Marcolino da Silva**

**PONTOALL:**

**Sistema Eletrônico de Controle de Ponto e Gestão de Jornada de Trabalho**

**Projeto Integrador**

Jales

2025

**Lemuel Prates Sobrinho Ferreira**

**Maria Clara Borges Cardoso**

**Vinicius Lúcio Marcolino da Silva**

**PONTOALL:**

**Sistema Eletrônico de Controle de Ponto e Gestão de Jornada de Trabalho**

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 Sistema.

Orientador: Prof. Jefferson Antônio Ribeiro Passarini.

Jales

2025

**Lemuel Prates Sobrinho Ferreira**

**Maria Clara Borges Cardoso**

**Vinicius Lúcio Marcolino da Silva**

**PONTOALL:**

**Sistema Eletrônico de Controle de Ponto e Gestão de Jornada de Trabalho**

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 Sistema.

Orientador: Prof. Jefferson Antônio Ribeiro Passarini.

**Banca Examinadora:**

Prof. (Orientador)

Instituição

Prof.

Instituição

Prof.

Instituição

Jales, de de .

Dedicatória

**AGRADECIMENTOS**

´

**RESUMO**

**ABSTRACT**

**LISTA DE FIGURAS**

[Figura 1 — Ilustração Ponto Manual 15](#_Toc192450392)

[Figura 2 — Relógio de Ponto Biométrico Biopoint II-S 15](#_Toc192450393)

[Figura 3 — Telas da Plataforma Registro Ponto 16](#_Toc192450394)

[Figura 4 — Diagrama de Classes 12](#_Toc192450395)

**LISTA DE QUADROS**

[Quadro 1 — Requisitos Funcionais do Sistema 17](#_Toc192450346)

[Quadro 2 — Requisitos Não Funcionais do Sistema 20](#_Toc192450347)

[Quadro 3 — Descrição Classe User](#_Toc192450348) 25

[Quadro 4 — Descrição Classe Company](#_Toc192450349) 25

[Quadro 5 — Descrição Enum Company Status — Enumeração dos status da empresa](#_Toc192450350) 26

[Quadro 6 — Descrição Classe Face Embeddings](#_Toc192450351) 26

[Quadro 7 — Descrição Classe Scale](#_Toc192450352) 27

[Quadro 8 — Descrição Enum Day Type — Enumeração dos tipos de dias](#_Toc192450353) 28

[Quadro 9 — Descrição Classe Validation Area](#_Toc192450354) 28

[Quadro 10 — Descrição Classe Geo Location](#_Toc192450355) 28

[Quadro 11 — Descrição Enum User Status — Enumeração dos status dos usuários](#_Toc192450356) 29

[Quadro 12 — Descrição Enum User Type — Enumeração dos tipos dos usuários](#_Toc192450357) 29

[Quadro 13 — Descrição Classe Workload](#_Toc192450358) 30

[Quadro 14 — Descrição Classe Report](#_Toc192450359) 30

[Quadro 15 — Descrição Classe Mark Point](#_Toc192450360) 31

[Quadro 16 — Descrição Classe Department](#_Toc192450360) 31

[Quadro 17 — Descrição Classe Sector](#_Toc192450360) 31

**LISTA DE ABREVIATURAS E SIGLAS**

**CLT** - Consolidação das Leis do Trabalho

**UML** - *Unified Modeling Language*

**SUMÁRIO**

[**1 INTRODUÇÃO** 11](#_Toc192503772)

[**2 LEVANTAMENTO DE REQUISITOS DE SOFTWARE** 13](#_Toc192503773)

[2.1 Descrição dos Objetivos do Sistema 13](#_Toc192503774)

[2.2 Análise de Sistemas Existentes 14](#_Toc192503775)

[2.3 Descrição dos Principais Problemas 16](#_Toc192503776)

[2.4 Descrição dos Requisitos Funcionais 17](#_Toc192503777)

[2.5 Descrição dos Requisitos Não Funcionais 19](#_Toc192503778)

[**3 VISÃO DE CASO DE USO — UML** 11](#_Toc192503779)

[3.1 Diagrama de Classes 11](#_Toc192503780)

[3.2 Dicionário de Classes 13](#_Toc192503781)

[**REFERÊNCIAS** 11](#_Toc192503782)

# **1 INTRODUÇÃO**

O controle da jornada de trabalho tem sido um tema amplamente discutido atualmente, especialmente com o aumento da digitalização dos processos empresariais. O Ministério do Trabalho enfatiza que o registro de ponto não apenas visa evitar desavenças e ações judiciais relacionadas ao cumprimento dos direitos trabalhistas, mas também assegura a transparência e a precisão no controle das horas trabalhadas. Seu principal objetivo é garantir a segurança tanto para a empresa quanto para os trabalhadores, estabelecendo uma base sólida para o cálculo de salários e a verificação do cumprimento das normas legais (Moranho, 2024).

De acordo com a Consolidação das Leis do Trabalho (CLT) (Brasil, 2011), o controle de ponto é uma obrigação para empresas com mais de 10 funcionários, com o intuito de regular as horas trabalhadas, além de possibilitar o pagamento de horas extras e o cumprimento de intervalos. No entanto, as empresas devem atentar-se para os métodos adotados, já que cada um oferece uma taxa diferente de efetividade e segurança no processo de registro. Sistemas tradicionais, como o ponto manual ou mecânico, podem ser suscetíveis a falhas e fraudes, o que torna o uso de tecnologias mais avançadas, como o ponto eletrônico, uma escolha cada vez mais relevante para o cumprimento das normativas trabalhistas (JusBrasil, 2022).

É indiscutível a versatilidade que as medidas digitais têm trazido para o mercado de trabalho, especialmente no contexto da transformação digital que tem impactado diversos setores. O uso de sistemas digitais não só otimiza o processo de controle de jornada, mas também oferece vantagens como o armazenamento em tempo real dos registros e a possibilidade de acesso remoto. De acordo com o Ministério do Trabalho, a regulamentação da Portaria 373/2011, em 2011, foi um marco importante, permitindo a adoção de alternativas como o Ponto Eletrônico Online. Esse tipo de sistema digital proporciona praticidade, evitando a burocracia dos métodos tradicionais e tornando o processo mais ágil e preciso. Além disso, sua utilização assegura a conformidade com as leis trabalhistas, contribuindo para a redução de erros e fraudes, ao mesmo tempo que aumenta a transparência na gestão da jornada de trabalho (Moranho, 2024).

O problema que motiva este estudo é a necessidade de melhorar a segurança, inclusão e a eficácia dos sistemas de controle de jornada, evitando falhas que podem comprometer tanto os direitos trabalhistas quanto a operação das empresas. Diante disso, o objetivo deste trabalho é o desenvolvimento do software PontoAll, usado para o registro de ponto, que integra tecnologias de biometria facial, oferecendo um método seguro, eficiente e adaptável a diferentes modelos de jornada de trabalho.

Ao longo deste estudo, serão detalhadas as funcionalidades do sistema proposto, incluindo os requisitos técnicos, as tecnologias utilizadas e o processo de implementação.

# **2 LEVANTAMENTO DE REQUISITOS DE SOFTWARE**

O levantamento de requisitos constitui a etapa inicial do desenvolvimento de sistemas. Nessa fase, são discutidos os elementos que estarão presentes no sistema, bem como suas respectivas funcionalidades, elaborando e especificando ideias propostas, com foco na solução de problemas. Segundo Pressman:

Para estimular uma abordagem mais colaborativa e orientada a equipes em relação ao levantamento de requisitos, os envolvidos trabalham juntos para identificar o problema, propor elementos da solução, negociar diferentes abordagens e especificar um conjunto preliminar de requisitos da solução (Pressman, 2021, p. 110).

É importante destacar que os requisitos de um sistema são descrições e restrições do que o cliente deseja no sistema (Sommerville, 2019) Por esse motivo, seu levantamento é um processo complexo, pois exige uma compreensão aprofundada das metas de negócio para que o objetivo final do cliente seja atendido de maneira eficiente (Pressman, 2021).

## 2.1 Descrição dos Objetivos do Sistema

O principal objetivo do sistema é garantir um registro de ponto acessível, seguro e eficiente para todos os trabalhadores, independentemente de suas condições físicas ou do modelo de jornada adotado. Para isso, busca-se promover a inclusão, permitindo que pessoas com diferentes limitações, como impossibilidade de utilizar biometria digital, consigam registrar sua jornada de forma adequada e sem barreiras.

Além da acessibilidade, o sistema visa aprimorar a segurança e a confiabilidade dos registros de ponto, reduzindo fraudes e inconsistências nos dados. Um controle mais preciso das horas trabalhadas contribui para a transparência nas relações trabalhistas, evitando equívocos no cálculo de jornada, horas extras e faltas.

Outro objetivo é otimizar a gestão de informações dos funcionários, facilitando a organização e o acompanhamento da jornada de trabalho. Isso possibilita um melhor cumprimento das normas trabalhistas, garantindo que empresas e colaboradores tenham acesso a registros precisos e confiáveis.

Por fim, o sistema busca minimizar riscos de conflitos trabalhistas e ações judiciais decorrentes de erros no controle de ponto. Ao garantir um processo claro e bem documentado, a solução contribui para um ambiente corporativo mais seguro, justo e alinhado às exigências legais.

## 2.2 Análise de Sistemas Existentes

O controle de ponto é um componente fundamental na gestão de recursos humanos e na garantia do cumprimento das normas trabalhistas nas empresas. Ele assegura a transparência na contabilização das horas trabalhadas, evitando desavenças relacionadas a discrepâncias no horário de trabalho. Contudo, muitos métodos tradicionais de marcação de ponto, como cartões manuais, livros ou planilhas, se mostraram obsoletos, não atendendo mais às demandas de flexibilidade e acessibilidade exigidas pelo atual cenário do mercado de trabalho. Esses métodos não só demandam tempo e esforço humano, como também são propensos a erros, o que prejudica tanto os colaboradores quanto a gestão administrativa. A crescente adoção de tecnologias digitais tem proporcionado soluções mais eficientes, como os sistemas automatizados de controle de ponto, que integram as marcações diretamente às folhas de pagamento e permitem a atualização em tempo real. Esses sistemas trazem uma série de benefícios, como maior agilidade, precisão e redução de erros, facilitando a organização dos processos administrativos e a conformidade com a legislação vigente (Kl Quartz, 2024).

A transição para sistemas digitais também ajuda a minimizar problemas recorrentes nos modelos tradicionais, como a marcação de ponto indevida por colegas, o tempo excessivo dedicado à organização de planilhas e a dificuldade de acesso aos registros, que são comuns nos pontos digitais sem suporte adequado. Além disso, sistemas antigos e ineficazes frequentemente apresentam falhas, como a lentidão no carregamento de arquivos, o que impacta diretamente na produtividade da equipe e no tempo de resposta das informações. Dessa maneira, a adoção de tecnologias mais avançadas no controle de ponto é essencial para as empresas que buscam otimizar seus processos e garantir uma gestão mais eficiente e sem contratempos (Chies, 2024).

O ponto manual (Figura 1) é uma das formas tradicionais de controle de jornada de trabalho, em que o colaborador registra sua entrada e saída em um livro ou folha de ponto. Esse método exige que o funcionário anote manualmente os horários, o que pode resultar em erros de digitação, registros imprecisos ou até mesmo em fraudes, caso não haja uma supervisão rigorosa. Embora seja simples e de baixo custo, o ponto manual é suscetível a inconsistências e dificuldades na gestão de grandes volumes de dados. Além disso, não oferece a agilidade e a segurança de sistemas mais modernos, como os eletrônicos ou biométricos, que permitem um controle mais preciso e eficiente da jornada de trabalho (Cerqueira, 2023).

**Figura 1** ­­­­— Ilustração Ponto Manual



Fonte: Freepik Company S.L.

A Dimep Sistemas (Figura 2) é uma empresa especializada em soluções tecnológicas para controle de ponto, acesso, estacionamento e automação comercial. Entre as soluções oferecidas, destaca-se o relógio de ponto eletrônico, um dispositivo regulamentado que registra a jornada de trabalho de forma digital. Esse equipamento pode operar por meio de biometria, cartões de proximidade ou senhas para identificação dos funcionários. Além disso, possui a funcionalidade de gerar comprovantes impressos e integração com sistemas de gestão, possibilitando um controle de ponto mais eficiente e a automatização do cálculo da folha de pagamento, reduzindo erros e garantindo maior segurança nos registros.

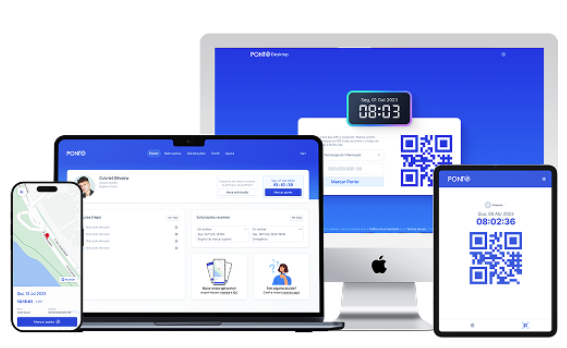
**Figura 2** — Relógio de Ponto Biométrico Biopoint II-S



Fonte: Dimep - Dimas De Melo Pimenta Sistemas De Ponto E Acesso.

Um sistema semelhante ao PontoAll é o Registro Ponto, uma plataforma online de controle de jornada de trabalho. Esse sistema permite que os colaboradores registrem seu ponto usando diversos dispositivos, como *smartphones*, *tablets* e computadores. Além disso, ele oferece diferentes opções de marcação, como o uso de biometria digital e facial, garantindo maior flexibilidade e segurança tanto para os funcionários quanto para a empresa.

**Figura 3** — Telas da Plataforma Registro Ponto



Fonte: Registro Ponto.

## 2.3 Descrição dos Principais Problemas

O sistema PontoAll foi desenvolvido com o objetivo de atender às necessidades das empresas no controle de entrada e saída de funcionários. Um dos principais problemas que ele busca resolver é a dificuldade de acessibilidade e complexidade que muitos sistemas tradicionais de controle de ponto apresentam. Frequentemente, esses sistemas são difíceis de usar ou não são adequados para todos os perfis de usuários, o que pode resultar em falhas operacionais e baixa adesão dos colaboradores. O PontoAll visa simplificar o processo de registro de ponto, tornando-o mais acessível e fácil de utilizar, independentemente da familiaridade dos funcionários com tecnologia.

Outro problema recorrente nos métodos tradicionais é a confusão na organização e processamento das informações coletadas, o que pode levar a erros no registro das jornadas de trabalho. Muitos sistemas não possuem validações adequadas ou mecanismos de segurança, tornando os dados propensos a falhas e dificultando a conformidade com as exigências legais. O PontoAll resolve essa questão ao garantir a precisão e segurança dos registros, com funcionalidades de validação automática e controle mais rigoroso das informações.

Entretanto, como qualquer sistema, o PontoAll também pode apresentar desafios. A implementação de uma nova ferramenta de controle de ponto pode encontrar resistência por parte de funcionários ou gestores, especialmente se houver dificuldades de adaptação ao sistema ou falhas durante a integração com outros sistemas já existentes na empresa. Além disso, a dependência de uma plataforma digital pode ser um desafio em ambientes onde a infraestrutura de TI é limitada ou instável, o que pode afetar a confiabilidade do sistema. Portanto, é fundamental que a empresa esteja preparada para essas possíveis dificuldades, oferecendo suporte técnico adequado e treinamento para garantir o uso efetivo da ferramenta.

## 2.4 Descrição dos Requisitos Funcionais

Os requisitos funcionais constituem um conjunto de especificações que descrevem o que o cliente espera que o software realize, detalhando suas funcionalidades de forma precisa. Essa etapa pode ser extremamente desafiadora, pois, em alguns casos, os clientes não possuem uma visão clara do que desejam do software. Por esse motivo, é fundamental garantir uma comunicação clara e eficaz entre as partes envolvidas no desenvolvimento do sistema (Guedes, 2011).

Existem duas formas principais de apresentação dos requisitos funcionais: os requisitos de usuário e os requisitos do sistema. Os requisitos de usuário são redigidos de maneira acessível, para que os diferentes tipos de usuários possam compreendê-los sem a necessidade de conhecimento técnico avançado. Eles descrevem as funcionalidades de forma a garantir que o sistema atenda às necessidades dos usuários finais, priorizando sua experiência e interação com o sistema. Por outro lado, os requisitos funcionais do sistema são mais técnicos e voltados para os desenvolvedores, abordando detalhadamente como as funcionalidades devem ser implementadas. Esses requisitos derivam dos requisitos de usuário e devem ser suficientemente claros para que a equipe de desenvolvimento saiba exatamente o que precisa ser feito em termos de código e arquitetura do sistema (Sommerville, 2011).

O sistema PontoAll foi desenvolvido para oferecer um controle eficiente da jornada de trabalho, assegurando segurança, transparência e conformidade com as normas trabalhistas. Para alcançar esses objetivos, foram estabelecidos requisitos funcionais específicos, detalhados no Quadro 1.

**Quadro 1** — Requisitos Funcionais do Sistema

|  |  |  |
| --- | --- | --- |
|  | Requisitos Funcionais | Descrição |
| 1 | Cadastro da Empresa | Permitir o registro inicial da organização no sistema, estabelecendo a base para o gerenciamento hierárquico. |
| 2 | Cadastro de Usuários | Facilitar a inclusão de funcionários e gestores, atribuindo permissões específicas conforme seus papéis. |
| 3 | Cadastro de Departamento | Realizar o cadastro de departamentos específicos da empresa, de modo que cada usuário possa ser vinculado a um determinado departamento. |
| 4 | Cadastro de Setor | Cadastrar os setores da empresa de forma individualizada, permitindo a associação de cada usuário ao setor correspondente às suas funções ou área de atuação. |
| 5 | Cadastro de Biometria Facial | Registrar a biometria facial dos funcionários para autenticação segura durante a marcação de ponto. |
| 6 | Configuração de Geolocalização | Definir áreas permitidas para marcação de ponto, garantindo que os registros ocorram apenas em locais autorizados. |
| 7 | Sistema de Escala de Trabalho | Permitir o registro e o gerenciamento da escala de trabalho de cada funcionário, bem como a definição de uma mesma escala para um setor ou departamento inteiro, de forma flexível, possibilitando edições e ajustes conforme necessário, a fim de garantir que as escalas possam ser facilmente adaptadas às necessidades da empresa e às variações no quadro de funcionários. Além disso, os horários definidos na escala deverão corresponder aos horários em que os funcionários deverão realizar a marcação de ponto, assegurando a coerência entre a jornada planejada e os registros de frequência. |
| 8 | Marcação de Ponto com Reconhecimento Facial | Permitir que os funcionários registrem seu ponto através de reconhecimento facial, assegurando autenticidade e prevenindo fraudes. |
| 9 | Verificação de Localização na Marcação de Ponto | Confirmar que o registro de ponto ocorre dentro das áreas permitidas previamente configuradas, garantindo conformidade com as políticas da empresa. |
| 10 | Armazenamento de Informações Detalhadas de Marcação | Registrar dados como data e horário de cada marcação, permitindo um acompanhamento preciso e histórico completo das jornadas de trabalho. |
| 11 | Justificativa para Marcação Fora do Padrão | Exigir que, em casos de registros fora do horário ou área permitida, o funcionário forneça uma justificativa, que será analisada e validada pelo gestor responsável. |
| 12 | Notificações Automáticas para Lembrete de Marcação | Enviar lembretes automáticos aos funcionários para evitar esquecimentos na marcação de ponto, promovendo conformidade e pontualidade. |
| 13 | Acesso aos Registros de Ponto | Proporcionar aos administradores a capacidade de visualizar registros de ponto, facilitando o monitoramento e a tomada de decisões. |
| 14 | Validação de Justificativas Apresentadas | Oferecer aos gestores ferramentas para analisar e validar as justificativas fornecidas pelos funcionários para marcações fora do padrão estabelecido. |
| 15 | Geração de Relatórios Semanais e Mensais | Criar relatórios detalhados sobre a jornada de trabalho, incluindo informações sobre atrasos, faltas e outras ocorrências, auxiliando na gestão e no cumprimento das normas trabalhistas. |
| 16 | Cálculos Automáticos de Carga Horária e Horas Extras | Calcular automaticamente a carga horária total, horas extras, faltas e períodos de jornada noturna, reduzindo erros manuais e garantindo precisão nos pagamentos e conformidade legal. |
| 17 | Sistema de Login e Autenticação | Implementar um mecanismo de login que permita aos usuários acessarem o sistema de forma segura, utilizando credenciais únicas, como nome de usuário e senha, garantindo a autenticação adequada e o controle de acesso. |
| 18 | Recuperação de Senha | Fornecer uma funcionalidade que permita aos usuários recuperarem ou redefinirem suas senhas de acesso em caso de esquecimento ou perda, garantindo a continuidade do acesso ao sistema. |
| 19 | Edição de Dados dos Funcionários | Permitir que administradores ou gestores autorizados possam editar informações cadastrais dos funcionários, como dados pessoais, cargos e departamentos, assegurando a atualização e precisão dos registros. |
| 20 | Edição de Horários de Ponto | Permitir a alteração dos horários de registro de ponto diretamente na escala, garantindo flexibilidade na gestão das jornadas de trabalho. |
| 21 | Edição de Configurações de geolocalização | Permitir a alteração das configurações de geolocalização, possibilitando ajustes conforme as necessidades operacionais ou alterações nas políticas da empresa. |
| 22 | Auditoria e Log de Atividades | Manter registros detalhados de todas as atividades e alterações realizadas no sistema, permitindo auditorias e rastreamento de ações para garantir a conformidade e identificar possíveis irregularidades. |
| 23 | Aplicativo Móvel para Funcionários | Disponibilizar um aplicativo móvel que permita aos funcionários registrar o ponto e acompanhar sua carga horária total, incluindo horas extras e faltas, de maneira prática e transparente. |
| 24 | Emissão de Comprovante de Registro de Ponto | Fornecer ao funcionário um comprovante impresso ou digital a cada marcação de ponto, conforme exigido pela legislação vigente. |

Fonte: Elaborado pelos autores.

## 2.5 Descrição dos Requisitos Não Funcionais

Requisitos não funcionais são especificações que se referem a aspectos gerais do sistema, não diretamente ligados aos serviços oferecidos, mas ao seu comportamento como um todo. Eles estão relacionados a restrições, condições, consistência e validações que complementam os requisitos funcionais (Guedes, 2011).

Segundo Pressman (2021), os requisitos não funcionais podem ser descritos como atributos ou características do sistema que envolvem qualidade, desempenho, segurança ou outras limitações que não estão diretamente ligadas às funcionalidades principais do sistema. Esses requisitos são essenciais para garantir que o sistema não apenas execute suas funções básicas, mas também atenda a aspectos importantes como usabilidade, confiabilidade e eficiência.

A negligência dos requisitos não funcionais pode resultar em sérios problemas no funcionamento do sistema, podendo torná-lo inviável ou até mesmo inutilizável. Isso ocorre porque, embora os requisitos funcionais estejam focados nas necessidades específicas do usuário, os requisitos não funcionais garantem que o sistema funcione adequadamente em termos de desempenho e segurança. Por outro lado, enquanto os requisitos funcionais podem ser alterados ou ajustados para atender às necessidades do usuário, os requisitos não funcionais, se desconsiderados, podem comprometer toda a infraestrutura do sistema (Sommerville, 2011).

Os requisitos não funcionais, apresentados no Quadro 2, foram cuidadosamente elaborados para assegurar que o sistema PontoAll não apenas atenda às funcionalidades essenciais, mas também opere de maneira eficiente, segura e intuitiva. Esses requisitos abrangem aspectos cruciais como desempenho, segurança, usabilidade, disponibilidade, acessibilidade, manutenibilidade, compatibilidade, interoperabilidade e as tecnologias empregadas.

**Quadro 2** — Requisitos Não Funcionais do Sistema

|  |  |  |
| --- | --- | --- |
|  | Requisitos Não Funcionais | Descrição |
| 1 | Tempo de Resposta | O sistema deve processar a autenticação do usuário via reconhecimento facial em menos de 5 segundos. |
| 2 | Tempo de Resposta da Web API | O tempo de resposta da Web API deve ser inferior a 1 segundo para operações comuns. |
| 3 | Verificação de Localização | Definir áreas permitidas para marcação de ponto, garantindo que os registros ocorram apenas em locais autorizados. |
| 4 | Precisão da Geolocalização | A precisão da geolocalização deve ser de, no mínimo, 10 metros de erro máximo. |
| 5 | Desempenho do Aplicativo Android | O aplicativo Android deve operar de forma fluida em dispositivos com no mínimo 2GB de RAM. |
| 6 | Proteção contra Tentativas de Fraude | O sistema deve possuir proteção contra tentativas de fraude, como o uso de fotos ou autenticação por meio de terceiros. |
| 7 | Autenticação e Autorização da API | A API deve utilizar JWT (JSON Web Token) para autenticação e autorização dos usuários. |
| 8 | Armazenamento de Dados Biométricos | Os dados biométricos devem ser armazenados de forma criptografada. |
| 9 | Comunicação Segura | A comunicação entre o frontend, backend e o aplicativo deve ser protegida via SSL/TLS 1.3. |
| 10 | Feedback Visual e Auditivo | Deve oferecer feedback visual e auditivo para indicar sucesso ou falha no registro. |
| 11 | Design da Interface | O design deve ser simples e minimalista, facilitando a navegação. |
| 12 | Consistência no Design | Deve utilizar padrões de design consistentes em todas as telas. |
| 13 | Acesso às Ações Principais | As ações principais devem estar visíveis e de fácil acesso na interface. |
| 14 | Monitoramento Ativo | Deve possuir um monitoramento ativo para detectar falhas rapidamente. |
| 15 | Backup Automático | O sistema deve possuir um backup automático dos registros. |
| 16 | Taxa de Erro do Reconhecimento Facial | O reconhecimento facial deve permitir uma taxa de erro máxima de 5%, garantindo precisão e rapidez. |
| 17 | Disponibilidade do Sistema | O sistema deve estar disponível 24 horas por dia, 7 dias por semana, salvo períodos de manutenção programada. |
| 18 | Compatibilidade com Tecnologias Assistivas | Deve ser compatível com leitores de tela e permitir ajustes de contraste e tamanho de fonte. |
| 19 | Comandos por Voz e Navegação Simplificada | Deve oferecer comandos por voz e suporte para navegação simplificada para as pessoas com deficiência motora. |
| 20 | Configuração de Alto Contraste | O sistema deve permitir a configuração de alto contraste para usuários com dificuldades visuais. |
| 21 | Confirmação Pós-Registro de Ponto | O usuário deve receber confirmação visual e sonora após registrar ponto. |
| 22 | Conformidade com Diretrizes de Acessibilidade | A interface web deve ser compatível com diretrizes de acessibilidade WCAG 2.1, permitindo o uso por pessoas com deficiência visual. |
| 23 | Documentação e Comentários | Utilização de comentários claros e consistentes e documentação técnica para novos desenvolvedores. |
| 24 | Modularidade do Sistema | O sistema deve ser modular, permitindo a atualização ou substituição de componentes sem impactar o funcionamento geral. |
| 25 | Testes em Ambiente de Staging | As atualizações devem ser testadas em ambientes de staging antes de serem implementadas no ambiente de produção, garantindo a qualidade do sistema. |
| 26 | Monitoramento Contínuo | O sistema deve ter um monitoramento contínuo, coletando logs detalhados sobre o desempenho, erros e falhas para análise e correção de problemas. |
| 27 | Compatibilidade com Android | O sistema deve ser compatível com Android 10+. |
| 28 | Responsividade do Aplicativo | O aplicativo deve ser responsivo e adaptável a diferentes tamanhos de tela. |
| 29 | Exportação de Relatórios | O sistema deve conseguir exportar relatórios no formato JSON/XML para dados complexos ou CSV para dados simples. |
| 30 | Tecnologias do Frontend Web | O frontend web será construído em React com Tailwind CSS, garantindo alta performance e responsividade. |
| 31 | Diretrizes de Design | As interfaces visuais devem seguir as diretrizes definidas pela Material Design 3, incluindo seus ícones, Material Icons. |
| 32 | Tecnologias da Web API | A Web API será desenvolvida em C# com .NET 8 e Entity Framework Core. |

Fonte: Elaborado pelos autores.

# **3 VISÃO DE CASO DE USO — UML**

De acordo com Pressman (2021), a UML (*Unified Modeling Language*) é uma linguagem de modelagem padronizada utilizada para descrever, especificar, construir e documentar projetos de software. Ela oferece uma abordagem visual para a representação das características presentes em um sistema, sendo amplamente adotada no desenvolvimento de software. Esse padrão facilita a comunicação entre os membros da equipe de desenvolvimento, proporcionando uma documentação clara e detalhada das diversas partes do sistema.

Seu principal foco é modelar softwares baseados na orientação a objetos. Seu objetivo é auxiliar os engenheiros de software na elaboração de sistemas como um todo, abrangendo aspectos como a definição de características, o comportamento dos componentes, as estruturas lógicas e a dinâmica dos processos. Além disso, pode ser utilizada para representar até mesmo necessidades físicas do sistema (Guedes, 2011).

## 3.1 Diagrama de Classes

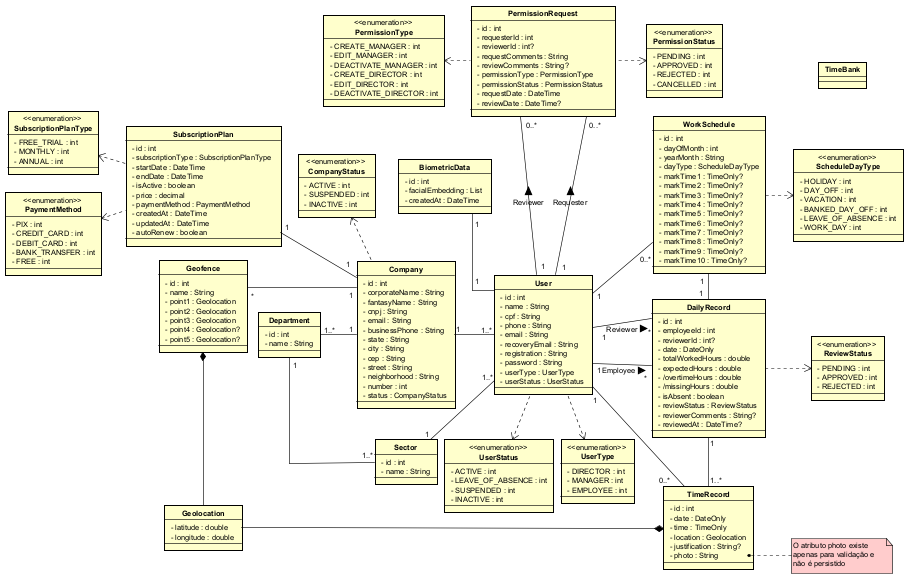
O diagrama de classes apresenta uma visão estática ou estrutural do sistema, sendo utilizado para a modelagem das classes dentro do contexto da UML. Esse tipo de diagrama descreve detalhadamente as classes do sistema, incluindo seus atributos, operações e as relações de associação com outras classes, proporcionando uma visão clara da estrutura do sistema a ser desenvolvido (Pressman, 2021).

Considerado um dos diagramas mais importantes e amplamente utilizados na UML, o diagrama de classes desempenha um papel fundamental na documentação e no planejamento de sistemas orientados a objetos. Entre seus principais objetivos, destaca-se a capacidade de ilustrar os componentes essenciais das classes, como seus atributos (propriedades ou características) e métodos (operações ou comportamentos), além de evidenciar como essas classes se inter-relacionam dentro do sistema. O diagrama de classes não apenas auxilia na visualização e compreensão do design do sistema, mas também serve como base para a construção de outros diagramas, como os diagramas de sequência, de atividades e de estados, entre outros (Guedes, 2011).

Uma classe pode ser encarada como um modelo para a criação de objetos dentro do sistema. À medida que a implementação é desenvolvida, os objetos são instanciados a partir dessas classes, representando os dados manipulados e garantindo a organização da lógica do software. Segundo Sommerville (2019), essa abordagem permite que os sistemas sejam estruturados de maneira modular, favorecendo a reutilização de código, a manutenção e a escalabilidade.

A Figura 4 apresenta o diagrama de classe, que oferece uma visão estrutural do sistema PontoAll. Ele ilustra as principais classes envolvidas no funcionamento do sistema, seus atributos e métodos, bem como os relacionamentos entre elas. O diagrama permite entender como as classes interagem entre si, representando a organização interna do sistema e como os dados fluem entre os diferentes componentes.

**Figura 4** — Diagrama de Classes



Fonte: Elaborado pelos autores

## 3.2 Dicionário de Classes

Os dicionários de classes são um recurso essencial na modelagem de sistemas orientados a objetos, permitindo a documentação detalhada das classes, atributos, métodos e seus relacionamentos. Eles auxiliam na definição precisa dos elementos que compõem o sistema, promovendo uma visão clara da estrutura e do comportamento dos objetos. Além disso, são fundamentais para garantir a padronização e a consistência do desenvolvimento, servindo como referência para todas as fases do projeto. Dessa forma, contribuem diretamente para a organização do código e facilitam a comunicação entre os membros da equipe, melhorando a eficiência do desenvolvimento e a manutenção do software (Sommerville, 2018).

A classe “BiometricData” (Quadro 3) armazena informações biométricas dos usuários. Ela inclui um identificador único, o embedding facial para reconhecimento, e a data e hora de criação do registro. Essa estrutura permite a validação e autenticação biométrica no sistema.

**Quadro 3** — Descrição Classe BiometricData

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único do registro biométrico |
| facialEmbedding | List | Representação numérica do rosto para reconhecimento facial |
| createdAt | DateTime | Data e hora em que os dados biométricos foram registrados |

Fonte: Elaborado pelos autores

A classe “Company” (Quadro 4) é responsável por representar as empresas cadastradas no sistema. Cada instância contém informações essenciais para a identificação e o contato com a empresa, incluindo razão social, nome fantasia, CNPJ, e-mail, telefone comercial, além de dados de localização como estado, cidade, CEP, rua, bairro e número. Também inclui o status da empresa, que determina se ela está ativa, inativa ou suspensa, conforme a política de assinaturas do sistema.

**Quadro 4** — Descrição Classe Company

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| Id | Integer | Identificador único da empresa |
| corporateName | String | Nome empresarial registrado (razão social) |
| fantasyName | String | Nome fantasia utilizado pela empresa |
| cnpj | String | Cadastro Nacional de Pessoa Jurídica |
| email | String | E-mail de contato principal da empresa |
| businessPhone | String | Número de telefone comercial da empresa |
| state | String | Unidade federativa onde a empresa está localizada |
| city | String | Cidade em que a empresa está localizada |
| cep | String | Código de Endereçamento Postal |
| street | String | Nome da rua onde a empresa está situada |
| neighborhood | String | Bairro onde a empresa está localizada |
| number | String | Número do edifício da empresa |
| status | CompanyStatus | Status da empresa no sistema (ativa, inativa ou suspensa) |

Fonte: Elaborado pelos autores

O enumerador “CompanyStatus” (Quadro 5) define os estados possíveis de uma empresa no sistema. Essa informação é usada para controlar o acesso e funcionamento da conta da empresa de acordo com sua situação administrativa, especialmente relacionada à assinatura de serviços.

**Quadro 5** — Descrição Enum CompanyStatus — Enumeração dos status da empresa

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| ACTIVE | Integer | Indica que a empresa está com a conta ativa e funcional |
| SUSPENDED | Integer | Indica que a empresa está temporariamente suspensa |
| INACTIVE | Integer | Indica que a empresa está inativa e não possui acesso ao sistema |

Fonte: Elaborado pelos autores

A classe “DailyRecord” (Quadro 6) representa o resumo diário das marcações de ponto de um funcionário. Ela registra os horários totais trabalhados, a expectativa de horas para o dia, possíveis ausências, status da revisão e comentários adicionais do revisor. É por meio dessa classe que se organiza e valida a jornada de trabalho diária dos colaboradores, permitindo controle e acompanhamento da assiduidade e pontualidade.

**Quadro 6** — Descrição Classe DailyRecord

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único do registro diário |
| employeeId | Integer | Identificador do funcionário ao qual o registro pertence |
| reviewerId | Integer? | Identificador do usuário que realizou a revisão |
| date | DateOnly | Data correspondente ao registro |
| totalWorkedHours | Double | Total de horas efetivamente trabalhadas pelo funcionário |
| expectedHours | Double | Quantidade de horas previstas para o dia |
| /overtimeHours | Double | Quantidade de horas trabalhadas além do previsto |
| /missingHours | Double | Quantidade de horas não cumpridas em relação ao esperado |
| isAbsent | Boolean | Indica se o funcionário esteve ausente |
| reviewStatus | ReviewStatus | Situação da revisão do registro (pendente, aprovado, ajustado ou rejeitado) |
| reviewerComments | String? | Comentários inseridos durante a revisão |
| reviewedAt | DateTime? | Data e hora em que a revisão foi concluída |

Fonte: Elaborado pelos autores

A classe “Department” (Quadro 7) representa uma divisão organizacional dentro de uma empresa. É utilizada para estruturar os setores administrativos ou operacionais da empresa, facilitando a organização e o gerenciamento de usuários, setores e escalas dentro do sistema.

**Quadro 7** — Descrição Classe Department

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único do departamento |
| name | String | Nome do departamento |

Fonte: Elaborado pelos autores

A classe “Geofence” (Quadro 8) define uma área geográfica virtual associada a uma empresa, utilizada para validar se a marcação de ponto de um funcionário ocorre dentro da área permitida. Essa cerca é estabelecida pelos gerentes e pode ser composta por até cinco pontos geográficos que formam seu perímetro.

**Quadro 8** — Descrição Classe Geofence

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único da cerca virtual |
| name | String | Nome atribuído à cerca virtual |
| point1 | Geolocation | Primeiro ponto geográfico obrigatório da cerca |
| point2 | Geolocation | Segundo ponto geográfico obrigatório da cerca |
| point3 | Geolocation | Terceiro ponto geográfico obrigatório da cerca |
| point4 | Geolocation? | Quarto ponto geográfico opcional da cerca |
| point5 | Geolocation? | Quinto ponto geográfico opcional da cerca |

Fonte: Elaborado pelos autores

A classe “Geolocation” (Quadro 9) representa as coordenadas geográficas utilizadas para validar a posição de marcação de ponto dos usuários ou definir os limites de uma cerca virtual. Dentro dessa classe, os atributos *Latitude* e *Longitude* são responsáveis por armazenar as coordenadas de um ponto específico na Terra. A *Latitude* indica a posição no eixo vertical, representando a distância em relação à linha do equador, enquanto a *Longitude* marca a posição no eixo horizontal, indicando a distância em relação ao meridiano de Greenwich. Dessa forma, a classe encapsula esses dados de forma independente e reutilizável no sistema, permitindo determinar com precisão a localização geográfica e definir restrições de área.

**Quadro 9** — Descrição Classe Geolocation

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| latitude | Double | Coordenada de latitude da localização geográfica |
| longitude | Double | Coordenada de longitude da localização geográfica |

Fonte: Elaborado pelos autores

O enumerador “PaymentMethod” (Quadro 10) define os métodos de pagamento aceitos no sistema para a contratação de planos de assinatura pelas empresas clientes. Os valores cobrem formas de pagamento eletrônicas, tradicionais e a gratuidade para planos promocionais ou de teste.

**Quadro 10** — Descrição Enum PaymentMethod — Enumeração dos métodos de pagamento

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| PIX | Integer | Pagamento via sistema instantâneo brasileiro (PIX) |
| CREDIT\_CARD | Integer | Pagamento realizado com cartão de crédito |
| DEBIT\_CARD | Integer | Pagamento realizado com cartão de débito |
| BANK\_TRANSFER | Integer | Pagamento por meio de transferência bancária tradicional |
| FREE | Integer | Plano gratuito, sem cobrança de valor |

Fonte: Elaborado pelos autores

A classe “PermissionRequest” (Quadro 11) representa uma solicitação de permissão feita por um gerente para realizar ações restritas no sistema, como criação, edição ou desativação de usuários de nível igual ou superior. Ela registra tanto os dados do solicitante e revisor quanto os comentários e status do processo de avaliação.

**Quadro 11** — Descrição Classe PermissionRequest

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único da solicitação |
| requesterId | Integer | ID do usuário que realizou a solicitação |
| reviewerId | Integer? | ID do diretor responsável pela revisão da solicitação, se aplicável |
| requestComments | String | Comentários ou justificativa inseridos pelo solicitante |
| reviewComments | String? | Comentários do revisor ao aprovar ou rejeitar a solicitação |
| permissionType | PermissionType | Tipo de permissão solicitada |
| permissionStatus | PermissionStatus | Estado atual da solicitação (pendente, aprovada, rejeitada, etc.) |
| requestDate | DateTime | Data e hora em que a solicitação foi criada |
| reviewDate | DateTime? | Data e hora em que a solicitação foi analisada, se aplicável |

Fonte: Elaborado pelos autores

O enumerador “PermissionStatus” (Quadro 12) define os possíveis estados de uma solicitação de permissão no sistema. Ele é utilizado para acompanhar o progresso da solicitação desde o momento em que é criada até sua finalização, seja por aprovação, rejeição ou cancelamento.

**Quadro 12** — Descrição Enum PermissionStatus — Enumeração dos tipos de estados de uma solicitação

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| PENDING | Integer | Solicitação foi enviada, mas ainda não foi analisada |
| APPROVED | Integer | Solicitação foi revisada e aprovada |
| REJECTED | Integer | Solicitação foi revisada e recusada |
| CANCELLED | Integer | Solicitação foi cancelada antes de ser analisada |

Fonte: Elaborado pelos autores

O enumerador “PermissionType” (Quadro 13) representa os tipos de permissões administrativas que podem ser solicitadas no sistema. Cada valor corresponde a uma ação específica que afeta usuários com papel de gerente ou diretor, como criação, edição ou desativação.

**Quadro 13** — Descrição Enum PermissionType — Enumeração dos tipos de permissões administrativas

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| CREATE\_MANAGER | Integer | Solicitação para criar um novo gerente |
| EDIT\_MANAGER | Integer | Solicitação para editar os dados de um gerente existente |
| DEACTIVATE\_MANAGER | Integer | Solicitação para desativar um gerente |
| CREATE\_DIRECTOR | Integer | Solicitação para criar um novo diretor |
| EDIT\_DIRECTOR | Integer | Solicitação para editar os dados de um diretor existente |
| DEACTIVATE\_DIRECTOR | Integer | Solicitação para desativar um diretor |

Fonte: Elaborado pelos autores

O enumerador “ReviewStatus” (Quadro 14) define os possíveis estados de uma revisão das marcações de um funcionário no sistema. Ele inclui os estados pendente, ajustado, aprovado e rejeitado. Esta estrutura padroniza o fluxo de revisão, garantindo que o status de cada item seja consistentemente representado.

**Quadro 14** — Descrição Enum ReviewStatus — Enumeração dos tipos de estados de uma revisão

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| PENDING | Integer | Indica que a revisão está aguardando avaliação |
| APPROVED | Integer | Indica que a revisão foi aceita |
| REJECTED | Integer | Indica que a revisão foi recusada |

Fonte: Elaborado pelos autores

O enumerador “ScheduleDayType” (Quadro 15) define os tipos de dias que um usuário pode ter em sua escala de trabalho. Ele inclui classificações como feriado, folga, férias, folga compensatória, licença e dia útil. Essa estrutura categoriza os dias para um gerenciamento preciso da jornada de trabalho.

**Quadro 15** — Descrição Enum ScheduleDayType — Enumeração dos tipos de dias

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| HOLIDAY | Integer | Indica que o dia é um feriado oficial |
| DAY\_OFF | Integer | Indica que o dia é uma folga regular |
| VACATION | Integer | Indica que o dia faz parte de um período de férias |
| BANKED\_DAY\_OFF | Integer | Indica que o dia é uma folga compensatória por horas extras |
| LEAVE\_OF\_ABSENCE | Integer | Indica que o dia é uma licença (médica, pessoal, etc.) |
| WORK\_DAY | Integer | Indica que o dia é um dia normal de trabalho |

Fonte: Elaborado pelos autores

A classe “Sector” (Quadro 16) representa um setor dentro da empresa. Ela contém informações básicas como o identificador único e o nome do setor. Essa estrutura permite organizar os funcionários e departamentos por suas respectivas áreas de atuação.

**Quadro 16** — Descrição Classe Sector

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único do setor |
| name | String | Nome do setor |

Fonte: Elaborado pelos autores

A classe “SubscriptionPlan” (Quadro 17) representa um plano de assinatura no sistema. Ela armazena detalhes como o tipo de assinatura, datas de início e término, status de atividade, preço, método de pagamento, e o registro de criação e atualização, além de indicar se a assinatura possui renovação automática. Essa estrutura permite gerenciar os diferentes planos de assinatura disponíveis para as empresas clientes.

**Quadro 17** — Descrição Classe SubscriptionPlan

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único do plano de assinatura |
| subscriptionType | SubscriptionPlanType | Tipo do plano de assinatura (mensal, anual, etc.) |
| startDate | DateTime | Data de início da assinatura |
| endDate | DateTime | Data de término da assinatura |
| isActive | Boolean | Indica se a assinatura está ativa |
| price | Decimal | Preço do plano de assinatura |
| paymentMethod | PaymentMethod | Método de pagamento utilizado para a assinatura |
| createdAt | DateTime | Data e hora de criação do registro da assinatura |
| updatedAt | DateTime | Data e hora da última atualização do registro da assinatura |
| autoRenew | Boolean | Indica se a assinatura será renovada automaticamente |

Fonte: Elaborado pelos autores

O enumerador “SubscriptionPlanType” (Quadro 18) define os tipos de planos de assinatura disponíveis no sistema. Ele inclui opções como teste gratuito, plano mensal e anual. Essa estrutura padroniza as categorias de planos, facilitando a gestão e oferta de serviços.

**Quadro 18** — Descrição Enum SubscriptionPlanType — Enumeração dos tipos de planos de assinatura

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| FREE\_TRIAL | Integer | Indica um período de teste gratuito da assinatura |
| MONTHLY | Integer | Indica um plano de assinatura mensal |
| ANNUAL | Integer | Indica um plano de assinatura anual |

Fonte: Elaborado pelos autores

A classe “TimeRecord” (Quadro 19) registra os detalhes de um registro de ponto. Ela inclui o identificador, a data, a hora e a localização do funcionário no momento do registro. O campo de justificativa é opcional, e o atributo *photo*, embora obrigatório para validação, não é persistido. Essa estrutura é essencial para o controle de frequência e gerenciamento de jornada de trabalho.

**Quadro 19** — Descrição Classe TimeRecord

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único do registro de ponto |
| date | DateOnly | Data em que o ponto foi registrado |
| time | TimeOnly | Hora em que o ponto foi registrado |
| location | Geolocation | Localização geográfica onde o ponto foi registrado |
| justification | String? | Justificativa para o registro do ponto, caso necessário |
| photo | String | Imagem utilizada para validação do ponto (não persistido) |

Fonte: Elaborado pelos autores

A classe “User” (Quadro 20) representa os atributos dos usuários do sistema. Ela inclui informações como nome, CPF, telefone, e-mail principal e de recuperação, senha, registro do funcionário na empresa, tipo do usuário e o status do usuário. Essa estrutura permite o gerenciamento dos usuários, garantindo a identificação e categorização conforme seu papel e situação na empresa.

**Quadro 20** — Descrição Classe User

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único do usuário |
| name | String | Nome completo do usuário |
| cpf | String | Cadastro de Pessoa Física do usuário |
| phone | String | Número de telefone do usuário |
| email | String | Endereço de e-mail principal do usuário |
| recoveryEmail | String | Endereço de e-mail secundário para recuperação de conta |
| registration | String | Matrícula do funcionário na empresa |
| password | String | Senha de acesso do usuário |
| userType | UserType | Tipo de usuário no sistema (diretor, gerente, funcionário) |
| userStatus | UserStatus | Estado atual do usuário (ativo, suspenso, etc.) |

Fonte: Elaborado pelos autores

O enumerador “UserStatus” (Quadro 21) define os possíveis estados de um usuário no sistema. Inclui classificações como ativo, licença, suspenso e inativo. Essa estrutura padroniza o status dos usuários, auxiliando no gerenciamento de acesso e controle.

**Quadro 21** — Descrição Enum UserStatus — Enumeração dos tipos de estado de um usuário

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| ACTIVE | Integer | Indica que o usuário está ativo no sistema |
| LEAVE\_OF\_ABSENCE | Integer | Indica que o usuário está de licença |
| SUSPENDED | Integer | Indica que o acesso do usuário foi suspenso |
| INACTIVE | Integer | Indica que o usuário está inativo no sistema |

Fonte: Elaborado pelos autores

O enumerador “UserType” (Quadro 22) define os diferentes perfis de usuários no sistema, determinando seus níveis de acesso e responsabilidades. O tipo *DIRECTOR* representa o usuário com maior autoridade, responsável por visualizar informações estratégicas e definir ações para os gerentes. O tipo *MANAGER* corresponde ao usuário que gerencia informações operacionais e define os horários dos funcionários. Já o tipo *EMPLOYEE* identifica o usuário com acesso restrito, encarregado somente da marcação do próprio ponto. Essa estrutura categoriza os usuários de acordo com seu papel e nível de acesso.

**Quadro 22** — Descrição Enum UserType— Enumeração dos tipos de perfis de um usuário

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| DIRECTOR | Integer | Indica um usuário com papel de diretor |
| MANAGER | Integer | Indica um usuário com papel de gerente |
| EMPLOYEE | Integer | Indica um usuário com papel de funcionário |

Fonte: Elaborado pelos autores

A classe “WorkSchedule” (Quadro 23) gerencia os horários de trabalho diários por funcionário. Ela registra o dia do mês, o mês e ano de referência, o tipo de dia (como dia útil ou feriado), e até dez horários de marcação de ponto. Essa estrutura é fundamental para organizar e monitorar a jornada de trabalho dos funcionários.

**Quadro 23** — Descrição Classe WorkSchedule

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| id | Integer | Identificador único do registro de horário de trabalho |
| dayOfMonth | Integer | O dia do mês para o qual o horário está definido |
| yearMonth | String | O mês e o ano de referência do horário ("YYYY-MM") |
| dayType | ScheduleDayType | O tipo de dia (dia útil, feriado, folga, etc.) |
| markTime1 | TimeOnly? | Primeiro horário de marcação de ponto do dia (opcional) |
| markTime2 | TimeOnly? | Segundo horário de marcação de ponto do dia (opcional) |
| markTime3 | TimeOnly? | Terceiro horário de marcação de ponto do dia (opcional) |
| markTime4 | TimeOnly? | Quarto horário de marcação de ponto do dia (opcional) |
| markTime5 | TimeOnly? | Quinto horário de marcação de ponto do dia (opcional) |
| markTime6 | TimeOnly? | Sexto horário de marcação de ponto do dia (opcional) |
| markTime7 | TimeOnly? | Sétimo horário de marcação de ponto do dia (opcional) |
| markTime8 | TimeOnly? | Oitavo horário de marcação de ponto do dia (opcional) |
| markTime9 | TimeOnly? | Nono horário de marcação de ponto do dia (opcional) |
| markTime10 | TimeOnly? | Décimo horário de marcação de ponto do dia (opcional) |

Fonte: Elaborado pelos autores

## 3.3 Definição dos Atores

O identificador que representa os objetos externos que interagem com os casos de uso, presentes em um Diagrama de Caso de Uso, é denominado ator. Cada papel desempenhado pelos usuários de um software é modelado por meio dos atores do sistema (Filho, 2009). Qualquer elemento externo que interaja com o sistema pode ser considerado um ator, incluindo eventuais representações de hardware ou software externos que estejam relacionados. Dessa maneira, a representação dos atores ocorre por meio de símbolos de “bonecos magros”, acompanhados de uma breve descrição abaixo, identificando seu papel dentro do diagrama (Guedes, 2011).

Na aplicação criada, foram definidos diferentes tipos de atores, cada um com funções e responsabilidades específicas, conforme sua posição hierárquica de acordo com o departamento de Recursos Humanos da empresa.

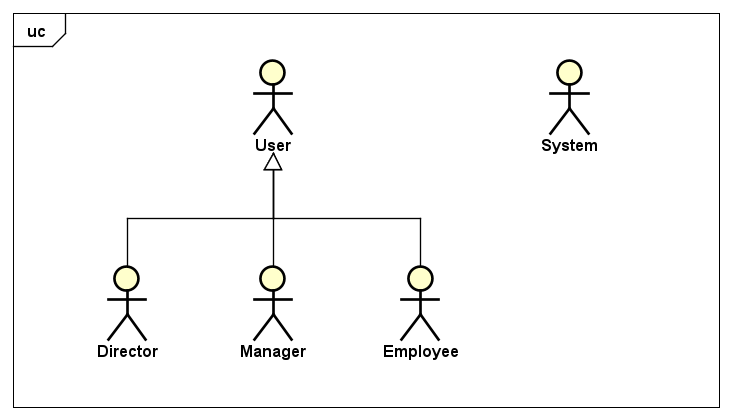
O Diretor (Director) ocupa o cargo de mais alto nível no sistema. Sua principal responsabilidade é validar as transações realizadas pelos demais atores. Além disso, possui acesso total à aplicação, podendo gerenciar todos os tipos de funcionários, escalas, departamentos, setores e cercas virtuais, bem como validar justificativas e gerar relatórios com base nas marcações de ponto. Esse conjunto de permissões permite que o Diretor visualize e controle amplamente as informações relacionadas à gestão de pessoas e à estrutura organizacional. Assim, sua função estratégica garante a supervisão global das operações administrativas.

O Gerente (Manager) também desempenha um papel essencial na administração do sistema, sendo responsável por gerenciar as atividades vinculadas aos funcionários. Seu perfil de acesso contempla as mesmas funcionalidades atribuídas ao Diretor; no entanto, as ações relativas ao cadastro de gerentes ou funcionários devem ser validadas pelo Diretor, assegurando maior controle e segurança nas operações.

Tanto o Diretor, quanto o Gerente e o Funcionário (Employee) possuem perfil de acesso à versão móvel do sistema, uma vez que todos são contratados e, portanto, devem realizar o registro de ponto. Por meio dessa interface, eles podem registrar sua jornada de trabalho, consultar suas escalas e visualizar o histórico de marcações. Esse modelo de acesso visa garantir simplicidade e objetividade na utilização das funcionalidades diretamente relacionadas às suas atividades diárias.

A Figura 5 apresenta o Diagrama de Atores, que fornece uma representação visual dos diferentes perfis de usuários identificados no sistema. Esse diagrama tem como objetivo ilustrar, de forma clara e objetiva, os atores envolvidos, destacando a existência e distinção entre os diversos tipos de usuários, sem detalhar fluxos de interação ou processos.

**Figura 5** ­­­­­­­­— Diagrama de Atores



Fonte: Elaborado pelos autores

## 3.4 Lista de Casos de Uso

Os casos de uso são empregados para a captura dos requisitos de um sistema, representando os serviços, tarefas ou funcionalidades identificadas como essenciais para o seu funcionamento. Esses casos servem para documentar e expressar os comportamentos e os diferentes tipos de interação que os atores podem estabelecer com o sistema. A sua representação ocorre por meio de elipses, nas quais é inserido um texto descritivo que identifica a funcionalidade correspondente (Guedes, 2011).

Com o objetivo de especificar e documentar cada caso de uso elaborado para o software PontoAll, foram estruturadas duas listas: a primeira corresponde à lista de mensagens do sistema, apresentada no Quadro 16; e a segunda refere-se à listagem dos casos de uso, apresentada no Quadro 17.

Para garantir a clareza na comunicação entre os atores e os casos de uso, foi elaborada uma lista de mensagens. Essa lista detalha os tipos de mensagens que podem ser exibidas, conforme as interações estabelecidas entre os usuários e os elementos do sistema. Esse recurso visa padronizar as respostas emitidas pelo sistema, assegurando maior consistência nas operações.

**Quadro 16** — Lista de Mensagens

|  |  |
| --- | --- |
| **Nº da Mensagem** | **Conteúdo da Mensagem** |
| Msg 1 | Login realizado com sucesso |
| Msg 2 | Recuperação de senha enviada com sucesso |
| Msg 3 | Transação validada com sucesso |
| Msg 4 | Transação reprovada com sucesso |
| Msg 5 | Cadastrado com sucesso |
| Msg 6 | Alterado com sucesso |
| Msg 7 | Excluído com sucesso |
| Msg 8 | Justificativa validada com sucesso |
| Msg 9 | Justificativa reprovada com sucesso |
| Msg 10 | Relatório gerado com sucesso |
| Msg 11 | Ponto registrado com sucesso |
| Msg 12 | Justificativa enviada com sucesso |
| Msg 13 | Email ou senha incorretos |
| Msg 14 | Campos obrigatórios não preenchidos |
| Msg 15 | Erro ao conectar com o banco |
| Msg 16 | Registro não encontrado |

Fonte: Elaborado pelos autores

Com o intuito de facilitar a compreensão das funcionalidades associadas a cada caso de uso que integra os diagramas, elaborou-se um quadro explicativo (Quadro 17). Nela, cada caso de uso foi devidamente enumerado, descrito de forma objetiva e acompanhado da indicação de suas respectivas entradas e saídas, proporcionando uma visão estruturada e sistemática dos processos representados. Essa abordagem visa também otimizar a implementação por parte dos desenvolvedores, promovendo maior clareza e eficiência no desenvolvimento do sistema.

**Quadro 17** — Lista de Casos de Uso

| **N°** | **Descrição do Caso de Uso** | **Entrada** | **Caso de Uso** | **Resposta** |
| --- | --- | --- | --- | --- |
| 1 | Diretor realiza login | Informa dados de acesso | Login | Msg 1 |
| 2 | Diretor recupera senha | Informa dados de acesso | Recuperar Senha | Msg 2 |
| 3 | Diretor solicita lista de transações | Sem entrada | Listar Transações | Retorna lista de transações |
| 4 | Diretor valida transação | Id transação | Validar Transação | Msg 3 |
| 5 | Diretor reprova transação | Id transação | Reprovar Transação | Msg 4 |
| 6 | Diretor solicita lista de funcionários | Sem entrada | Listar Funcionários | Retorna lista de funcionários |
| 7 | Diretor cadastra funcionário | Informa dados do funcionário | Cadastrar Funcionário | Msg 5 |
| 8 | Diretor edita dados do funcionário | Informa dados do funcionário | Editar Funcionário | Msg 6 |
| 9 | Diretor deleta funcionário | Id funcionário | Deletar Funcionário | Msg 7 |
| 10 | Diretor filtra funcionários por departamento | Id departamento | Filtrar por Departamento | Retorna lista filtrada |
| 11 | Diretor filtra funcionário por setor | Id setor | Filtrar por Setor | Retorna lista filtrada |
| 12 | Diretor solicita lista de escala dos funcionários | Sem entrada | Listar Escala | Retorna lista de escalas |
| 13 | Diretor cadastra escala do funcionário | Informa dados da escala | Cadastrar Escala | Msg 5 |
| 14 | Diretor edita dados da escala do funcionário | Informa dados da escala | Editar Escala | Msg 6 |
| 15 | Diretor deleta escala do funcionário | Id escala | Deletar Escala | Msg 7 |
| 16 | Diretor filtra escalas por departamento | Id departamento | Filtrar por Departamento | Retorna lista filtrada |
| 17 | Diretor filtra escalas por setor | Id setor | Filtrar por Setor | Retorna lista filtrada |
| 18 | Diretor solicita lista de departamentos | Sem entrada | Listar Departamento | Retorna lista de departamentos |
| 19 | Diretor cadastra departamento | Informa dados do departamento | Cadastrar Departamento | Msg 5 |
| 20 | Diretor edita dados de departamento | Informa dados do departamento | Editar Departamento | Msg 6 |
| 21 | Diretor deleta departamento | Id departamento | Deletar Departamento | Msg 7 |
| 22 | Diretor solicita lista de setores | Sem entrada | Lista Setor | Retorna lista de setores |
| 23 | Diretor cadastra setor | Informa dados do setor | Cadastrar Setor | Msg 5 |
| 24 | Diretor edita dados do setor | Informa dados do setor | Editar Setor | Msg 6 |
| 25 | Diretor deleta setor | Id setor | Deletar Setor | Msg 7 |
| 26 | Diretor solicita lista de cercas virtuais | Sem entrada | Listar Cercas Virtuais | Retorna lista de cercas virtuais |
| 27 | Diretor cadastra cerca virtual | Informa dados da cerca virtual | Cadastrar Cerca | Msg 5 |
| 28 | Diretor edita cerca virtual | Informa dados da cerca virtual | Editar Cerca | Msg 6 |
| 29 | Diretor deleta cerca virtual | Id cerca virtual | Deletar Cerca | Msg 7 |
| 30 | Diretor solicita lista de justificativas | Sem entrada | Listar Justificativas | Retorna lista de justificativas |
| 31 | Diretor reprova justificativa | Id justificativa | Reprovar Justificativa | Msg 9 |
| 32 | Diretor valida justificativa | Id justificativa | Validar Justificativa | Msg 8 |
| 33 | Diretor solicita lista de marcações de ponto | Sem entrada | Listar Marcações de Ponto | Retorna lista de marcações de ponto |
| 34 | Diretor filtra marcações por departamento | Id departamento | Filtrar por Departamento | Retorna lista filtrada |
| 35 | Diretor filtra marcações por setor | Id setor | Filtrar por Setor | Retorna lista filtrada |
| 36 | Diretor gera relatório | Id funcionário | Gerar Relatório | Msg 10 |
| 37 | Gerente realiza login | Informa dados de acesso | Login | Msg 1 |
| 38 | Gerente recupera senha | Informa dados de acesso | Recuperar Senha | Msg 2 |
| 39 | Gerente solicita lista de funcionários | Sem entrada | Listar Funcionários | Retorna lista de funcionários |
| 40 | Gerente cadastra funcionário | Informa dados do funcionário | Cadastrar Funcionário | Msg 5 |
| 41 | Gerente edita funcionário | Informa dados do funcionário | Editar Funcionário | Msg 6 |
| 42 | Gerente deleta funcionário | Id funcionário | Deletar Funcionário | Msg 7 |
| 43 | Gerente filtra funcionários por departamento | Id departamento | Filtrar por Departamento | Retorna lista filtrada |
| 44 | Gerente filtra funcionário por setor | Id setor | Filtrar por Setor | Retorna lista filtrada |
| 45 | Gerente solicita lista de escala dos funcionários | Sem entrada | Listar Escala | Retorna lista de escalas |
| 46 | Gerente cadastra escala do funcionário | Informa dados da escala | Cadastrar Escala | Msg 5 |
| 47 | Gerente edita dados da escala do funcionário | Informa dados da escala | Editar Escala | Msg 6 |
| 48 | Gerente deleta escala do funcionário | Id escala | Deletar Escala | Msg 7 |
| 49 | Gerente filtra escalas por departamento | Id departamento | Filtrar por Departamento | Retorna lista filtrada |
| 50 | Gerente filtra escalas por setor | Id setor | Filtrar por Setor | Retorna lista filtrada |
| 51 | Gerente solicita lista de departamentos | Sem entrada | Listar Departamento | Retorna lista de departamentos |
| 52 | Gerente cadastra departamento | Informa dados do departamento | Cadastrar Departamento | Msg 5 |
| 53 | Gerente edita dados de departamento | Informa dados do departamento | Editar Departamento | Msg 6 |
| 54 | Gerente deleta departamento | Id departamento | Deletar Departamento | Msg 7 |
| 55 | Gerente solicita lista de setores | Sem entrada | Lista Setor | Retorna lista de setores |
| 56 | Gerente cadastra setor | Informa dados do setor | Cadastrar Setor | Msg 5 |
| 57 | Gerente edita dados do setor | Informa dados do setor | Editar Setor | Msg 6 |
| 58 | Gerente deleta setor | Id setor | Deletar Setor | Msg 7 |
| 59 | Gerente solicita lista de cercas virtuais | Sem entrada | Listar Cercas Virtuais | Retorna lista de cercas virtuais |
| 60 | Gerente cadastra cerca virtual | Informa dados da cerca virtual | Cadastrar Cerca | Msg 5 |
| 61 | Gerente edita cerca virtual | Informa dados da cerca virtual | Editar Cerca | Msg 6 |
| 62 | Gerente deleta cerca virtual | Id cerca virtual | Deletar Cerca | Msg 7 |
| 63 | Gerente solicita lista de justificativas | Sem entrada | Listar Justificativas | Retorna lista de justificativas |
| 64 | Gerente reprova justificativa | Id justificativa | Reprovar Justificativa | Msg 9 |
| 65 | Gerente valida justificativa | Id justificativa | Validar Justificativa | Msg 8 |
| 66 | Gerente solicita lista de marcações de ponto | Sem entrada | Listar Marcações de Ponto | Retorna lista de marcações de ponto |
| 67 | Gerente filtra marcações por departamento | Id departamento | Filtrar por Departamento | Retorna lista filtrada |
| 68 | Gerente filtra marcações por setor | Id setor | Filtrar por Setor | Retorna lista filtrada |
| 69 | Gerente gera relatório | Id funcionário | Gerar Relatório | Msg 10 |
| 70 | Usuário realiza login | Informa dados de acesso | Login | Msg 1 |
| 71 | Usuário recupera senha | Informa dados de acesso | Recuperar Senha | Msg 2 |
| 72 | Usuário visualiza escala | Id escala | Visualizar Escala | Retorna escala |
| 73 | Usuário bate ponto | Id marcação | Bater Ponto | Msg 11 |
| 74 | Usuário justifica atraso ou falta ao trabalho | Id marcação | Justificar | Msg 12 |
| 75 | Usuário solicita suas marcações de ponto | Sem entrada | Listar Marcações | Retorna lista de marcações de ponto |

Fonte: Elaborado pelos autores

## 3.5 Diagrama de Casos de Uso

No contexto da modelagem de requisitos funcionais, o Diagrama de Casos de Uso tem como principal finalidade representar uma visão abstrata das funcionalidades de um produto de software, evidenciando os relacionamentos existentes entre os casos de uso e os atores. Assim, os casos de uso descrevem funções específicas que o sistema deve desempenhar, enquanto os atores correspondem aos diferentes perfis de usuários ou sistemas externos que interagem com o software (Filho, 2009). Esse diagrama é considerado o mais abstrato, flexível e informal dentre os elementos da UML. Caracterizando-se como uma prática recomendada para as etapas iniciais do processo de modelagem, uma vez que fornece a base conceitual para a elaboração de outros diagramas. Ressalta-se, contudo, que ele pode sofrer ajustes ao longo do desenvolvimento, à medida que novos requisitos são identificados ou refinados (Guedes, 2011).

Rascunho: A Figura 6 apresenta o diagrama de caso de uso da visão do Diretor, onde mostra quais serão as interações do diretor com o sistema. O Diretor é um usuário com o cargo mais alto em relação ao nível hierárquico, pois é ele quem valida determinadas ações que os gerentes realizam no sistema, como cadastros de outros gerentes ou diretores. Além disso, o diretor tem acesso a todas as funcionalidades do sistema, como cadastro, alteração, exclusão, listagem em informações do sistema.

**Figura 6** — Diagrama de Caso de Uso - Visão Diretor

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

**Figura 7** — Diagrama de Caso de Uso - Visão Gerente

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

**Figura 8** — Diagrama de Caso de Uso - Visão Usuário

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

## 3.6 Diagrama de Casos de Uso Individual

## 3.6.1 Diagrama de Casos de Uso Individual — Listar Cercas Específicas

**Figura 9** — Diagrama de Casos de Uso Específico – Listar Cercas Virtuais

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

**Quadro 18** — Documentação do caso de uso Listar Cercas Virtuais

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC01 – Listar Cercas Virtuais** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso detalha as etapas que o ator Gerente executa para consultar as informações referentes às cercas virtuais previamente cadastradas no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema, além de existirem cercas virtuais previamente cadastradas. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de CercasVirtuais no sistema. |  | |
|  | 2. O Sistema exibe a página de Cercas Virtuais, apresentando a listagem de todas as cercas previamente cadastradas. | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma cerca cadastrada. | |
| **Cenário Alternativo – “Lista Vazia”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1. O Gerente acessa a página de CercasVirtuais no sistema. | |  |
|  | | 2. O sistema detecta que a lista de cercas está vazia. |
|  | | 3. O sistema exibe a mensagem: "Nenhuma cerca cadastrada”. |
| **Cenário de Exceção – “Falha de Carregamento”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1. O Gerente acessa a página de CercasVirtuais no sistema. | |  |
|  | | 2. Sistema tenta buscar a lista de cercas, mas ocorre um erro. |
|  | | 3. Sistema exibe uma mensagem de erro: "Não foi possível carregar as cercas virtuais. Tente novamente mais tarde." |

Fonte: Elaborado pelos autores

**Quadro 19** — Documentação do caso de uso Cadastrar Cerca

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC02 – Cadastrar Cerca** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso descreve detalhadamente as etapas realizadas pelo ator Gerente para cadastrar as informações de uma cerca virtual no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de CercasVirtuais no sistema. |  | |
|  | 2. O Sistema exibe a página de Cercas Virtuais, apresentando a listagem de todas as cercas previamente cadastradas. | |
| |  | | --- | | 3. O Gerente seleciona o botão "Cadastrar Nova Cerca". | |  | |
|  | 4. O sistema apresenta o mapa para definição da nova cerca virtual. | |
| 5. O Gerente define os pontos que delimitam a nova cerca virtual. |  | |
|  | 6. O sistema valida os dados. | |
|  | 7. O sistema persiste a cerca virtual no banco de dados. | |
|  | 8. O sistema exibe mensagem de sucesso e redireciona para a listagem atualizada. | |
| 9. O gerente visualiza a lista atualizada com a nova cerca. |  | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema. | |
| **Cenário Alternativo – “Dados Inválidos”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-5. Seguem iguais ao fluxo principal. | |  |
|  | | 6. O sistema valida os dados e identifica informações inválidas. |
| |  | | --- | |  | | | 7. O sistema exibe mensagens de erro específicas indicando os problemas encontrados no cadastro. |
| **Cenário de Exceção – “Falha de Salvamento”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-7. Seguem iguais ao fluxo principal. | |  |
|  | | 8. O sistema tenta salvar os dados no banco, mas ocorre uma falha. |
|  | | 9. O sistema exibe uma mensagem de erro: "Não foi possível concluir o cadastro da cerca virtual no momento. Tente novamente mais tarde." |

Fonte: Elaborado pelos autores

**Quadro 20** — Documentação do caso de uso Editar Cerca

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC03 – Editar Cerca** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso detalha as etapas executadas pelo ator Gerente para editar as informações de uma cerca virtual já cadastrada no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema, além de existirem cercas virtuais previamente cadastradas. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de CercasVirtuais no sistema. |  | |
|  | 2. O Sistema exibe a página de Cercas Virtuais, apresentando a listagem de todas as cercas previamente cadastradas. | |
| |  | | --- | | 3. O Gerente seleciona a cerca virtual que deseja editar. | |  | |
|  | 4. O sistema exibe o mapa com os pontos atuais da cerca virtual selecionada. | |
| 5. O Gerente altera os pontos definidos no mapa. |  | |
| 6. O Gerente submete o mapa de edição. |  | |
|  | 7. O sistema valida as informações atualizadas. | |
|  | 8. O sistema salva as alterações no banco de dados. | |
|  | 9. O sistema exibe mensagem de sucesso e retorna à listagem atualizada. | |
| 10. O Gerente visualiza a listagem atualizada com as alterações aplicadas. |  | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma cerca cadastrada. | |
| **Cenário Alternativo – “Dados Inválidos”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-5. Seguem iguais ao fluxo principal. | |  |
|  | | 6. O sistema valida os dados e identifica informações inválidas. |
| |  | | --- | |  | | | 7. O sistema exibe mensagens de erro específicas indicando os problemas encontrados. |
| **Cenário de Exceção – “Falha de Salvamento”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-7. Seguem iguais ao fluxo principal. | |  |
|  | | 8. O sistema tenta salvar os dados no banco, mas ocorre uma falha. |
|  | | 9. O sistema exibe uma mensagem de erro: "Não foi possível salvar as alterações no momento. Tente novamente mais tarde." |

Fonte: Elaborado pelos autores

**Quadro 21** — Documentação do caso de uso Deletar Cerca

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC04 – Deletar Cerca** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso detalha as etapas executadas pelo ator Gerente para deletar uma cerca virtual já cadastrada no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema, além de existirem cercas virtuais previamente cadastradas. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de CercasVirtuais no sistema. |  | |
|  | 2. O Sistema exibe a página de Cercas Virtuais, apresentando a listagem de todas as cercas previamente cadastradas. | |
| |  | | --- | | 3. O Gerente identifica e seleciona a cerca virtual que deseja deletar. | |  | |
|  | 4. O sistema exibe uma solicitação de confirmação de exclusão. | |
| 5. O Gerente confirma a exclusão da cerca virtual. |  | |
|  | 6. O sistema remove a cerca virtual selecionada do banco de dados. | |
|  | 7. O sistema exibe uma mensagem de sucesso informando que a cerca foi excluída com êxito. | |
| 8. O Gerente visualiza a listagem atualizada sem a cerca deletada. |  | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma cerca cadastrada. | |
| **Cenário Alternativo – “Exclusão Cancelada”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
| 5. O Gerente decide não confirmar a exclusão da cerca virtual. | |  |
| |  | | --- | |  | | | 6. O sistema cancela a operação e mantém a cerca virtual no banco de dados. |
| **Cenário de Exceção – “Falha de Exclusão”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-6. Seguem iguais ao fluxo principal. | |  |
|  | | 8. O sistema tenta excluir a cerca virtual, mas ocorre uma falha. |
|  | | 9. O sistema exibe uma mensagem de erro: "Não foi possível excluir a cerca virtual no momento. Tente novamente mais tarde." |

Fonte: Elaborado pelos autores

## 3.6.2 Diagrama de Casos de Uso Individual — Listar Justificativas

**Figura 10** — Diagrama de Casos de Uso Específico – Listar Justificativas

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

**Quadro 22** — Documentação do caso de uso Listar Justificativas

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC05 – Listar Justificativas** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso descreve as etapas realizadas pelo ator Gerente para consultar as informações referentes às justificativas dos funcionários no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema, além de existirem justificativas previamente cadastradas. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de Justificativas no sistema. |  | |
|  | 2. O sistema exibe a página de Justificativas, apresentando a listagem de todas as justificativas previamente cadastradas. | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma justificativa cadastrada. | |
| **Cenário Alternativo – “Lista Vazia”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1. O Gerente acessa a página de Justificativas no sistema. | |  |
|  | | 2. O sistema verifica e identifica que não existem justificativas cadastradas. |
|  | | 3. O sistema exibe uma mensagem informativa: "Nenhuma justificativa encontrada." e mantém a tela sem registros. |
| **Cenário de Exceção – “Falha de Carregamento”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1. O Gerente acessa a página de CercasVirtuais no sistema. | |  |
|  | | 2. Sistema tenta buscar a lista de justificativas, mas ocorre um erro. |
|  | | 3. Sistema exibe uma mensagem de erro: "Não foi possível carregar as justificativas. Tente novamente mais tarde." |

Fonte: Elaborado pelos autores

**Quadro 23** — Documentação do caso de uso Reprovar Justificativas

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC06 – Reprovar Justificativas** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso descreve as etapas realizadas pelo ator Gerente para reprovar alguma justificativa de um funcionário no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema, além de existirem justificativas previamente cadastradas. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de Justificativas no sistema. |  | |
|  | 2. O sistema exibe a página de Justificativas, apresentando a listagem de todas as justificativas previamente cadastradas. | |
| 3. O Gerente seleciona a justificativa que deseja reprovar. |  | |
| 4. O Gerente clica na opção “Reprovar”. |  | |
|  | 5. O sistema solicita a confirmação da reprovação e, opcionalmente, o motivo. | |
| 6. O Gerente confirma a reprovação (e, se solicitado, informa o motivo). |  | |
|  | 7. O sistema altera o status da justificativa para “Reprovada” e registra o motivo, se informado. | |
|  | 8. O sistema exibe uma mensagem de sucesso: “Justificativa reprovada com sucesso.” | |
| 9. O Gerente visualiza a listagem atualizada com o status alterado. |  | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma justificativa cadastrada. | |
| **Cenário Alternativo – “Reprovação Cancelada”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
|  | | 5. O sistema solicita a confirmação da reprovação. |
| 6. O Gerente opta por cancelar a operação. | |  |
|  | | 7. O sistema cancela a reprovação e mantém a justificativa com o status original. |
| 8. O Gerente visualiza que a justificativa não foi alterada. | |  |
| **Cenário de Exceção – “Falha de Carregamento”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-7. Seguem iguais ao fluxo principal. | |  |
|  | | 2. O sistema tenta atualizar o status da justificativa, mas ocorre uma falha. |
|  | | 3. O sistema exibe uma mensagem de erro: “Não foi possível reprovar a justificativa. Tente novamente mais tarde.” |

Fonte: Elaborado pelos autores

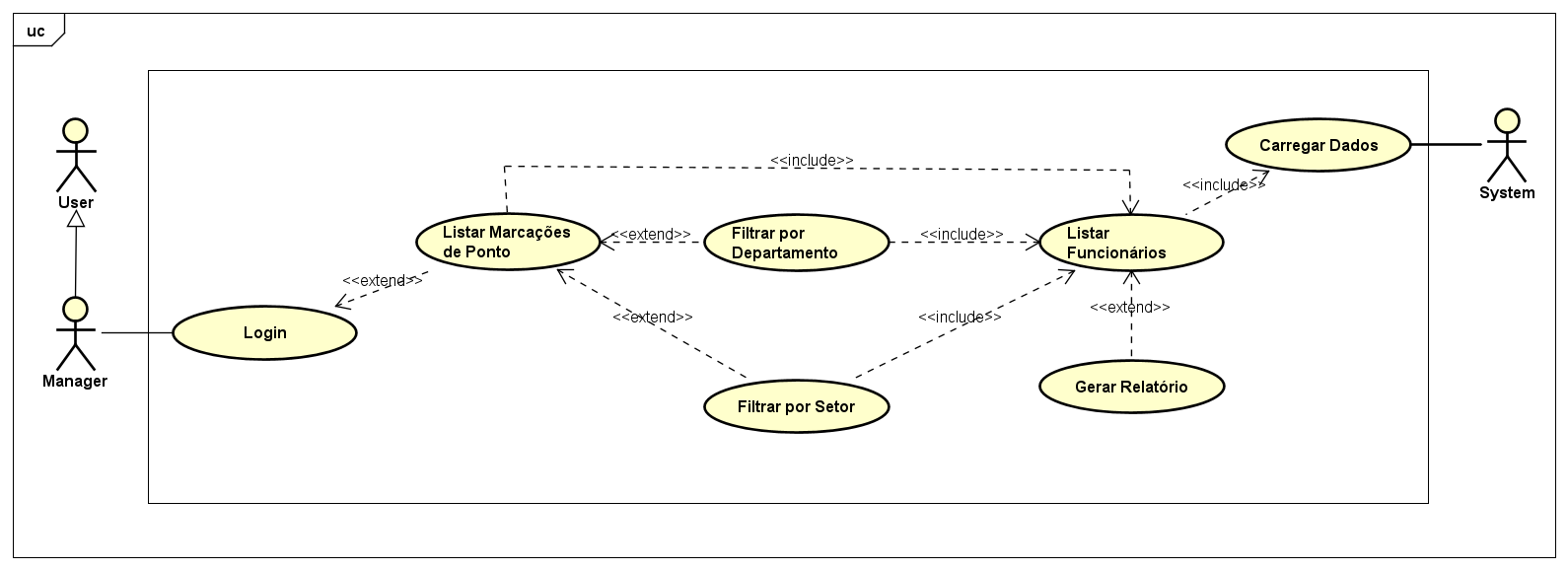
**Quadro 24** — Documentação do caso de uso Validar Justificativas

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC07 – Validar Justificativas** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso descreve as etapas realizadas pelo ator Gerente para validar alguma justificativa de um funcionário no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema, além de existirem justificativas previamente cadastradas. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de Justificativas no sistema. |  | |
|  | 2. O sistema exibe a página de Justificativas, apresentando a listagem de todas as justificativas previamente cadastradas. | |
| 3. O Gerente seleciona a justificativa que deseja validar. |  | |
| 4. O Gerente clica na opção “Validar”. |  | |
|  | 5. O sistema solicita a confirmação da validação. | |
| 6. O Gerente confirma a validação. |  | |
|  | 7. O sistema altera o status da justificativa para “Aprovada”. | |
|  | 8. O sistema exibe uma mensagem de sucesso: “Justificativa aprovada com sucesso.” | |
| 9. O Gerente visualiza a listagem atualizada com o status alterado. |  | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma justificativa cadastrada. | |
| **Cenário Alternativo – “Reprovação Cancelada”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
|  | | 5. O sistema solicita a confirmação da validação. |
| 6. O Gerente opta por cancelar a operação. | |  |
|  | | 7. O sistema cancela a validação e mantém a justificativa com o status original. |
| 8. O Gerente visualiza que a justificativa não foi alterada. | |  |
| **Cenário de Exceção – “Falha de Carregamento”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-7. Seguem iguais ao fluxo principal. | |  |
|  | | 2. O sistema tenta atualizar o status da justificativa, mas ocorre uma falha. |
|  | | 3. O sistema exibe uma mensagem de erro: “Não foi possível validar a justificativa. Tente novamente mais tarde.” |

Fonte: Elaborado pelos autores

## 3.6.3 Diagrama de Casos de Uso Individual — Listar Marcações de Ponto

**Figura 11** — Diagrama de Casos de Uso Específico – Listar Marcações de Ponto



Fonte: Elaborado pelos autores

**Quadro 25** — Documentação do caso de uso Listar Marcações de Ponto

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC08 – Listar Marcações de Ponto** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso descreve as etapas realizadas pelo ator Gerente para consultar as informações referentes às marcações de ponto dos funcionários no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema, além de existirem marcações de ponto previamente cadastradas. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de Marcações de Ponto no sistema. |  | |
|  | 2. O sistema exibe a página de Marcações de Ponto, apresentando a listagem de todas as marcações previamente cadastradas. | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma marcação de ponto cadastrada. | |
| **Cenário Alternativo – “Lista Vazia”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1. O Gerente acessa a página de Justificativas no sistema. | |  |
|  | | 2. O sistema verifica e identifica que não existem marcações de ponto cadastradas. |
|  | | 3. O sistema exibe uma mensagem informativa: "Nenhuma marcação de ponto encontrada." |
| **Cenário de Exceção – “Falha de Carregamento”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1. O Gerente acessa a página de CercasVirtuais no sistema. | |  |
|  | | 2. O sistema tenta buscar as marcações, mas ocorre uma falha. |
|  | | 3. Sistema exibe uma mensagem de erro: "Não foi possível carregar as marcações de ponto. Tente novamente mais tarde." |

Fonte: Elaborado pelos autores

**Quadro 26** — Documentação do caso de uso Filtrar por Departamento

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC09 – Filtrar por Departamento** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso descreve as etapas realizadas pelo ator Gerente para filtrar por departamento as informações referentes às marcações de ponto dos funcionários no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema. Além disso, é necessário que existam marcações de ponto e departamentos previamente cadastrados. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de Marcações de Ponto no sistema. |  | |
|  | 2. O sistema exibe a página de Marcações de Ponto, apresentando a listagem de todas as marcações previamente cadastradas. | |
| 3. O Gerente seleciona um departamento no campo de filtro disponível. |  | |
|  | 4. O sistema filtra as marcações de ponto, exibindo apenas aquelas relacionadas ao departamento selecionado. | |
| 5. O Gerente visualiza a listagem atualizada conforme o filtro aplicado. |  | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma marcação de ponto cadastrada.  3. Deve existir pelo menos um departamento cadastrado. | |
| **Cenário Alternativo – “Nenhuma Marcação Encontrada”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
|  | | 5. O sistema verifica que não há marcações de ponto associadas ao departamento selecionado. |
|  | | 6. O sistema exibe uma mensagem informativa: "Nenhuma marcação encontrada para o departamento selecionado." |
| **Cenário de Exceção – “Falha ao Aplicar Filtro”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
|  | | 2. O sistema tenta aplicar o filtro, mas ocorre uma falha. |
|  | | 3. O sistema exibe uma mensagem de erro: "Não foi possível aplicar o filtro. Tente novamente mais tarde." |

Fonte: Elaborado pelos autores

**Quadro 27** — Documentação do caso de uso Filtrar por Setor

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC10 – Filtrar por Setor** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso descreve as etapas realizadas pelo ator Gerente para filtrar por setor as informações referentes às marcações de ponto dos funcionários no sistema. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema. Além disso, é necessário que existam marcações de ponto e setores previamente cadastrados. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de Marcações de Ponto no sistema. |  | |
|  | 2. O sistema exibe a página de Marcações de Ponto, apresentando a listagem de todas as marcações previamente cadastradas. | |
| 3. O Gerente seleciona um setor no campo de filtro disponível. |  | |
|  | 4. O sistema filtra as marcações de ponto, exibindo apenas aquelas relacionadas ao setor selecionado. | |
| 5. O Gerente visualiza a listagem atualizada conforme o filtro aplicado. |  | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma marcação de ponto cadastrada.  3. Deve existir pelo menos um setor cadastrado. | |
| **Cenário Alternativo – “Nenhuma Marcação Encontrada”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
|  | | 5. O sistema verifica que não há marcações de ponto associadas ao setor selecionado. |
|  | | 6. O sistema exibe uma mensagem informativa: "Nenhuma marcação encontrada para o setor selecionado." |
| **Cenário de Exceção – “Falha ao Aplicar Filtro”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
|  | | 2. O sistema tenta aplicar o filtro, mas ocorre uma falha. |
|  | | 3. O sistema exibe uma mensagem de erro: "Não foi possível aplicar o filtro. Tente novamente mais tarde." |

Fonte: Elaborado pelos autores

**Quadro 28** — Documentação do caso de uso Gerar Relatório

|  |  |  |
| --- | --- | --- |
| **Nome do Caso de Uso** | **UC11 – Gerar Relatório** | |
| Ator Principal | Usuário (Gerente) | |
| Atores Secundários | Sistema | |
| Resumo | Este caso de uso descreve as etapas realizadas pelo ator Gerente para gerar relatórios baseados nas marcações de ponto de cada funcionário. | |
| Pré-condições | O gerente deve estar devidamente logado e autenticado no sistema. Além disso, é necessário que existam marcações de ponto previamente cadastradas. | |
| **Cenário Principal** | | |
| **Ações do Ator** | **Ações do Sistema** | |
| 1. O Gerente acessa a página de Marcações de Ponto no sistema. |  | |
|  | 2. O sistema exibe a página de Marcações de Ponto, apresentando a listagem de todas as marcações previamente cadastradas. | |
| 3. O Gerente seleciona os critérios de geração do relatório, como período, funcionário e setor. |  | |
| 4. O gerente confirma a geração do relatório. |  | |
|  | 5. O sistema processa as informações com base nos critérios selecionados. | |
|  | 6. O sistema gera o relatório e o disponibiliza para visualização ou download. | |
| |  | | --- | |  |  |  | | --- | | 7. O Gerente visualiza ou exporta o relatório gerado conforme sua necessidade. | |  | |
| Restrições/Validações | 1. O gerente deve estar logado no sistema.  2. Deve existir pelo menos uma marcação de ponto cadastrada. | |
| **Cenário Alternativo – “Nenhuma Marcação Encontrada”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
|  | | 5. O sistema processa as informações, mas verifica que não há marcações que correspondam aos critérios selecionados. |
|  | | 6. O sistema exibe uma mensagem informativa: **"Nenhum dado encontrado para os critérios selecionados."** |
| **Cenário de Exceção – “Falha na Geração do Relatório”** | | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1-4. Seguem iguais ao fluxo principal. | |  |
|  | | 2. O sistema tenta processar a geração do relatório, mas ocorre uma **falha**. |
|  | | 3. O sistema exibe uma mensagem de erro: **"Não foi possível gerar o relatório. Tente novamente mais tarde."** |

Fonte: Elaborado pelos autores

## 3.7 Diagrama de Sequência

## No contexto da UML, o Diagrama de Sequência é utilizado, principalmente, para representar as interações entre os atores e os objetos de um sistema, bem como as relações entre os próprios objetos. Esse tipo de diagrama tem como finalidade demonstrar, de forma temporal e ordenada, a sequência de mensagens trocadas entre os elementos durante a execução de um determinado caso de uso ou de uma instância específica desse caso (Sommerville, 2019).

A seguir, serão apresentados e descritos os principais elementos que compõem o diagrama de sequência, detalhando suas funções e responsabilidades dentro do fluxo de interação do sistema. Essa análise permite compreender como os componentes se comunicam entre si, evidenciando o papel de cada camada envolvida no processo.

* **Ator:** Representa o usuário que inicia a interação com a aplicação. Suas ações disparam os fluxos representados no diagrama.
* ***WebApp*:** Refere-se à interface do sistema acessada via navegador, responsável por capturar as ações do usuário e encaminhar as requisições para o backend da aplicação.
* ***Controller*:** Camada responsável por receber as requisições vindas da WebApp e direcionar os dados ao serviço correspondente.
* ***Service*:** Contém a lógica de negócio da aplicação. Realiza validações mais complexas, aplica regras específicas do domínio e coordena as operações necessárias para atender à solicitação do usuário.
* ***Repository*:** Camada intermediária entre a lógica de negócio e o banco de dados. É responsável por executar consultas, inserções, atualizações e exclusões de dados, encapsulando o acesso às estruturas persistidas.
* ***DBContext*:** Gerencia a conexão com o banco, o rastreamento de entidades e a persistência de alterações realizadas pelas operações.

## 3.7.1 Diagrama de Sequência - Geofence

Este diagrama ilustra o fluxo de interação entre os componentes do sistema durante o processo de cadastro de uma cerca virtual. O processo se inicia com o Manager acessando a funcionalidade de cercas virtuais na interface WebApp. Após selecionar a opção de cadastrar uma nova cerca e visualizar o mapa, o usuário marca os pontos desejados e confirma a operação. A requisição é então enviada ao GeofenceController, que repassa os dados para o GeofenceService. Este serviço aplica a lógica de negócio, instancia o objeto Geofence e solicita ao GeofenceRepository que persista o novo registro no banco de dados por meio do DBContext. Ao final, o sistema retorna uma confirmação de sucesso ao usuário.

**Figura 12** — Diagrama de Sequência - Geofence

Diagrama

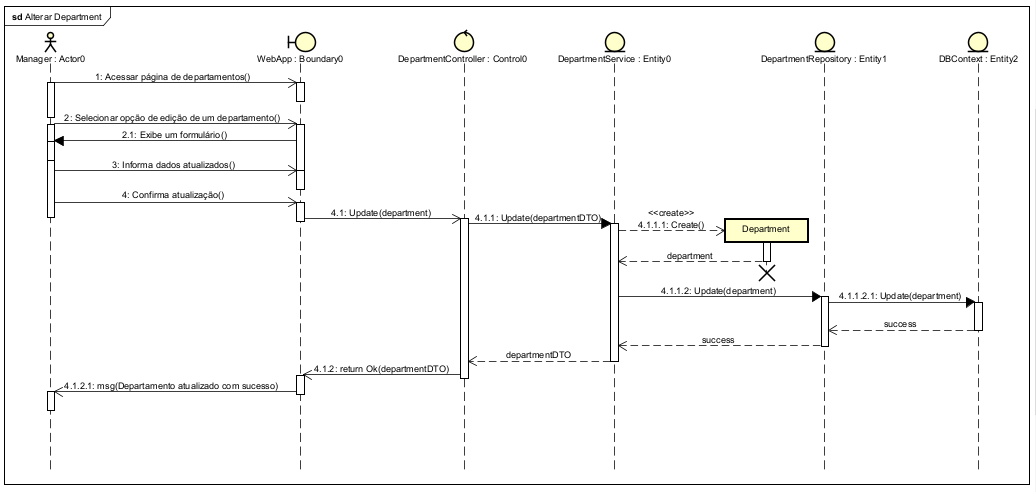
O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

## 3.7.2 Diagrama de Sequência - Alterar Department

Este diagrama descreve o fluxo de atualização de dados de um departamento no sistema. O processo tem início quando o Manager acessa a interface de departamentos via WebApp e seleciona a opção de editar um departamento. Um formulário é exibido, permitindo ao usuário visualizar e atualizar os dados atuais. Ao confirmar a atualização, a requisição é enviada ao DepartmentController, que a encaminha ao DepartmentService. Este serviço aplica a lógica de negócio, instancia o novo objeto Department com os dados atualizados e aciona o DepartmentRepository, que irá persistir a modificação no banco de dados através do DBContext. Ao término da operação, uma resposta de sucesso é retornada à interface, informando ao usuário que o departamento foi atualizado com sucesso.

**Figura 13** — Diagrama de Sequência - Alterar Department



Fonte: Elaborado pelos autores

## 3.7.3 Diagrama de Sequência - Listar User

Este diagrama apresenta o fluxo de consulta e listagem dos usuários do sistema. A sequência é iniciada pelo Manager, que acessa a página de funcionários via WebApp. A aplicação envia uma requisição ao UserController, que a repassa para o UserService. O serviço, por sua vez, solicita os dados ao UserRepository, que consulta o banco de dados por meio do DBContext. Os dados retornados são convertidos em um objeto UserDTO, representando uma versão simplificada dos usuários para exibição. O resultado final é devolvido ao *frontend*, que exibe a lista de funcionários ao usuário.

**Figura 14** — Diagrama de Sequência - Listar User

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

## 3.8 Diagrama de Máquina de Estado

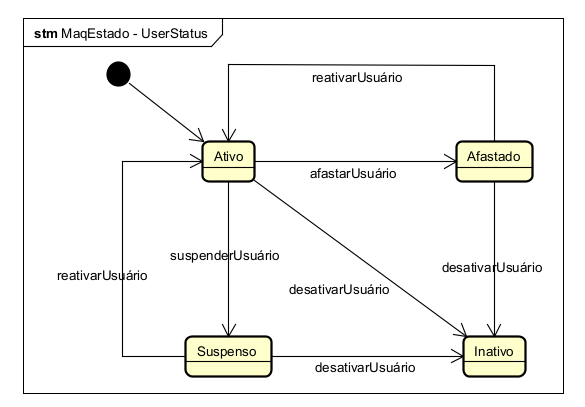
O Diagrama de Máquina de Estados é uma ferramenta da UML, utilizada para representar o comportamento de um elemento diante de um conjunto finito de transições de estado, denominado máquina de estados. Esse tipo de diagrama pode ser elaborado com base em um caso de uso, assim como o Diagrama de Sequência, ou aplicado para acompanhar o estado de outros elementos, como, por exemplo, a instância de uma classe (Guedes, 2011).

A máquina de estados consiste em um grafo formado por estados e transições. O estado representa uma condição ou situação durante o ciclo de vida de um objeto, enquanto a transição define a mudança de um estado para outro, indicando a relação entre essas condições (Filho, 2009).

3.8.1 Diagrama de Máquina de Estado – User Status

O objeto usuário pode apresentar os seguintes estados: Ativo, Afastado, Suspenso e Inativo. O estado inicial é Ativo. A partir dele, é possível realizar transições para os estados Afastado, por meio da ação afastarUsuário; Suspenso, por meio da ação suspenderUsuário; ou Inativo, por meio da ação desativarUsuário. Os estados Afastado e Suspenso permitem que retorne ao estado Ativo por meio da ação reativarUsuário. Contundo o estado Inativo representa uma condição final, sem transições de retorno.

**Figura 15** — Diagrama de Máquina de Estado - User Status

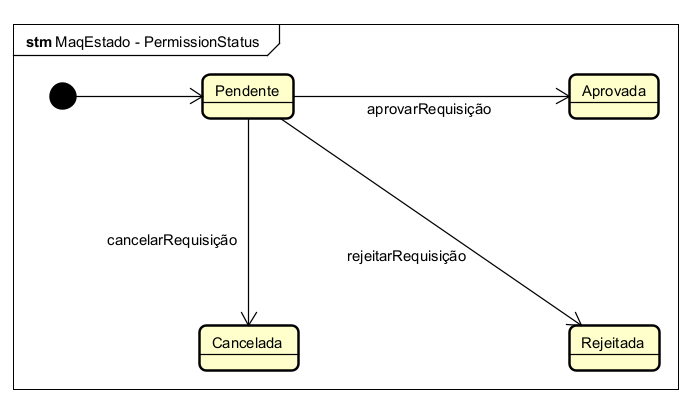


Fonte: Elaborado pelos autores

3.8.2 Diagrama de Máquina de Estado – Permission Status

O objeto PermissionStatus possui quatro estados possíveis: Pendente, Aprovada, Rejeitada e Cancelada, sendo Pendente o estado inicial. A partir desse estado, é possível realizar transições para Aprovada, através da ação aprovarRequisição; para Rejeitada, por meio da ação rejeitarRequisição; ou para Cancelada, utilizando a ação cancelarRequisição. Os estados Aprovada, Rejeitada e Cancelada são considerados finais, não permitindo novas transições a partir deles.

**Figura 16** — Diagrama de Máquina de Estado - Permission Status



Fonte: Elaborado pelos autores

3.8.3 Diagrama de Máquina de Estado – Company Status

O objeto CompanyStatus apresenta três estados possíveis: Ativa, Suspensa e Inativa. Ativa é o estado inicial. A partir dele, é possível realizar transições para Suspensa, por meio da ação suspenderEmpresa, ou para Inativa, por meio da ação desativarEmpresa. Do estado Suspensa, é possível retornar ao estado Ativa pela ação reativarEmpresa ou seguir para Inativa utilizando a ação desativarEmpresa. O estado Inativa representa uma condição final, não permitindo transições para outros estados.

**Figura 17** — Diagrama de Máquina de Estado - Company Status

Diagrama

O conteúdo gerado por IA pode estar incorreto.

# **4 DESCRIÇÃO DA INTERFACE COM O USUÁRIO (UX)**

A *User Experience* (UX), ou experiência do usuário, é compreendida como a criação e a integração de elementos que influenciam a experiência percebida durante a interação com uma aplicação. Seu objetivo é afetar positivamente as percepções, emoções e comportamentos do usuário ao longo desse processo. Para que um projeto seja desenvolvido de forma bem-sucedida, o design da experiência do usuário deve considerar tanto os objetivos de negócio quanto as necessidades específicas dos usuários da aplicação, além das limitações técnicas e operacionais que possam comprometer a viabilidade das funcionalidades propostas (Unger; Chandler, 2009).

## 4.1 Descrição de Cenário

Introdução

A Figura 18 ilustra a rotina de uma funcionária de uma fábrica de produção de couro ao realizar a marcação de ponto por meio do aplicativo móvel do PontoAll. O processo é simples, ágil e seguro, proporcionando uma experiência eficiente e confiável para o registro da jornada de trabalho.

**Figura 18** — Cenário 1 - Marcação de Ponto Segura no Ambiente Industrial

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

A Figura 19 apresenta a rotina de um gerente de Recursos Humanos responsável por cadastrar os horários de trabalho dos funcionários no sistema PontoAll. Por meio da definição de picks de horários previamente configurados, o processo de marcação de ponto se torna possível, promovendo organização, flexibilidade e controle sobre a jornada dos colaboradores.

**Figura 19** — Cenário 2 - Flexibilidade e Controle com Horários Pré-Definidos

Texto

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

## 4.2 Descrição de Persona

As personas são definidas como representações fictícias, que descrevem usuários típicos de uma determinada aplicação. Elas são úteis para a equipe de desenvolvimento, clientes e demais *stakeholders*, pois fornecem uma visão detalhada dos perfis dos usuários que irão interagir com o sistema, incluindo seus comportamentos, necessidades e objetivos. *Personas* bem elaboradas funcionam, em geral, como referência durante o processo de tomada de decisão, especialmente quando surgem dúvidas sobre como determinados aspectos do projeto devem ser estruturados ou implementados (Unger; Chandler, 2009).

Em específico para o PontoAll, duas personas foram elaboradas para melhor entendimento dos objetivos do projeto. A Persona 1, representada na Figura 20, é uma trabalhadora do setor industrial, atuando especificamente em uma fábrica de produção de couro localizada no interior do estado. Com uma rotina estruturada e disciplinada, demonstra comprometimento com suas atividades laborais e pontualidade em seu cotidiano profissional. Embora possua conhecimento limitado em tecnologias avançadas, apresenta facilidade no uso de aplicativos móveis para funções básicas do dia a dia, valorizando soluções que ofereçam praticidade, clareza e confiabilidade na execução de suas tarefas rotineiras.

**Figura 20** — Persona Operadora de Produção

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

A Persona 2, representada na Figura 21, é um profissional com cargo de gestão na área de Recursos Humanos, atuando em uma rede de supermercados atacadistas de médio a grande porte. Com ampla experiência na administração de pessoal, é responsável pela coordenação de escalas, supervisão de processos internos e garantia do cumprimento da jornada de trabalho dos colaboradores. Possui domínio de ferramentas digitais e demonstra constante interesse por soluções que otimizem o tempo, promovam organização e proporcionem maior controle sobre as atividades administrativas sob sua responsabilidade.

**Figura 21** — Persona Gerente de Recursos Humanos

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

## 4.3 Esboço de Tela (*wireframes*)

Os *wireframes* são representações visuais simplificadas das interfaces de uma aplicação, utilizadas para estruturar a disposição dos elementos em uma página ou tela. Esses protótipos de baixa fidelidade têm como principal finalidade definir a hierarquia da informação, os componentes funcionais e a organização do conteúdo, antes da aplicação de aspectos visuais mais detalhados. De acordo com Unger e Chandler (2009), os *wireframes* permitem identificar, de forma objetiva, os elementos que serão incorporados na interface, contribuindo para a validação inicial do layout e facilitando a comunicação entre os membros da equipe de desenvolvimento, designers e demais envolvidos no projeto.

**Figura 22** — Wireframe Tela de Login

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

A Figura 22 apresenta o *wireframe* da tela de login do sistema PontoAll. Os elementos foram organizados de forma a oferecer uma interface intuitiva, com o objetivo de facilitar a usabilidade e tornar a navegação simples e autoexplicativa. À direita da tela, estão dispostos os campos de login, o link para redefinição de senha e uma caixa de seleção para salvar os dados de acesso. Abaixo desses elementos, encontra-se o botão “Login”, a qual ser acionado, permite o acesso ao sistema. À esquerda da interface, está posicionada a logomarca do sistema, reforçando a identidade visual da aplicação.

**Figura 23** — Wireframe Tela Listagem de Funcionários

Tabela

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

A Figura 23 apresenta o *wireframe* da tela de listagem de funcionários, desenvolvido com o objetivo de proporcionar clareza visual e eficiência na navegação. A estrutura da interface é composta por uma barra lateral de navegação (*sidebar*), localizada à esquerda da tela, que permite o acesso às demais funcionalidades do sistema. Na parte superior, encontra-se o *header*, seguido pelo título da seção, posicionado logo abaixo, com a finalidade de identificar de forma clara o conteúdo exibido.

Na região central superior da tela, está localizada a barra de pesquisa, que facilita a localização de registros específicos na base de dados. Próximo a ela, encontra-se o botão de cadastro, que possibilita a inserção de novos funcionários ao sistema. A área principal da interface é ocupada pela tabela de listagem, que organiza e apresenta os dados dos funcionários cadastrados de forma estruturada e acessível. Cada linha da tabela dispõe de botões de edição e exclusão, permitindo ao usuário alterar ou remover registros conforme necessário.

**Figura 24** — Wireframe Modal de Cadastro Funcionário



Fonte: Elaborado pelos autores

A Figura 24 ilustra o *wireframe* da janela modal destinada ao cadastro de funcionários. A escolha por utilizar uma modal fundamentou-se na praticidade proporcionada por esse componente, que permite realizar o cadastro de forma rápida e eficiente. A organização dos campos segue uma estrutura lógica e de fácil compreensão, oferecendo ao usuário a possibilidade de cancelar a operação por meio do botão “Cancelar” ou confirmar o registro por meio do botão “Cadastrar”.

De maneira geral, os *wireframes* elaborados mantêm a padronização dos elementos utilizados, contribuindo para a consistência visual e funcional do sistema como um todo. Essa uniformidade na disposição dos componentes, como campos, botões, tabelas e menus, reforça a integridade da interface e facilita a navegação do usuário, garantindo uma experiência fluida e intuitiva em todas as telas da aplicação PontoAll. Além disso, a utilização de estruturas repetidas e bem definidas ao longo das diferentes interfaces permite que os usuários reconheçam padrões de interação, reduzindo a curva de aprendizado e aumentando a eficiência no uso do sistema.

## 4.4 Protótipos de Tela

No contexto da UX, a prototipação refere-se ao processo de criação e experimentação de partes das funcionalidades e elementos da aplicação em colaboração com os usuários, com o objetivo de avaliar e aperfeiçoar a interface e a experiência de uso (Unger; Chandler, 2009). Nesse processo, o protótipo consiste em uma representação visual e funcional do sistema, que permite aos *stakeholders* analisar sua estrutura, validar ideias e antecipar possíveis ajustes antes do desenvolvimento final (Rogers; Sharp; Preece, 2013).

Na aplicação PontoAll, a prototipação desempenhou um papel fundamental na comunicação entre os membros da equipe de projeto. Por meio dela, foi possível organizar ideias de forma clara, tomar decisões embasadas, desenvolver a identidade visual da aplicação e realizar testes de usabilidade. Esse processo contribuiu significativamente para a construção de uma solução coerente com as necessidades dos usuários e os objetivos do sistema.

**Figura 25** — Protótipo Tela de Login

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

A Figura 25 apresenta o protótipo da tela de login do sistema PontoAll. Na elaboração deste protótipo, foram seguidas as diretrizes propostas por Unger e Chandler (2009), com foco na forma como os usuários interagem com os elementos da interface. Por esse motivo, a disposição dos componentes, como campos de texto e botões, foi organizada verticalmente, respeitando a ordem natural de preenchimento, o que contribui para uma navegação fluida e intuitiva. A mesma lógica foi aplicada ao posicionamento do link para redefinição de senha e da caixa de seleção destinada à opção de lembrar os dados de acesso. À esquerda da tela, está localizada a área dedicada à identidade visual do sistema, na qual a logomarca é exibida juntamente com uma frase que reforça os objetivos centrais do PontoAll, promovendo o reconhecimento e a coerência visual da aplicação.

**Figura 26** — Protótipo Tela Listar Funcionários

Tabela

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

A tela de listagem de funcionários, apresentada na Figura 26, demonstra a organização visual dos elementos que compõem essa interface. A barra lateral de navegação (sidebar) está posicionada à esquerda do protótipo, mantendo a padronização adotada nas demais telas da aplicação, e apresenta a logomarca em versão reduzida na parte superior. O cabeçalho (header) está localizado na porção superior da tela, seguindo os princípios de design previamente estabelecidos. Logo abaixo, encontra-se o componente de título, que possui uma cor distinta do plano de fundo, a fim de destacar sua presença. Acima do título, está posicionado o breadcrumb, que contribui para a orientação do usuário dentro do sistema.

O conteúdo principal da página é representado pela tabela de listagem, acompanhada de suas respectivas funcionalidades. Na parte superior da tabela, há uma seção que inclui a barra de pesquisa, utilizada para localizar funcionários específicos, e o botão de cadastro, destinado à inserção de novos registros. A tabela em si é organizada por meio de colunas, sendo a primeira composta por caixas de seleção (checkboxes) que permitem selecionar linhas inteiras. As colunas seguintes exibem os dados dos funcionários, com destaque para a primeira coluna de conteúdo, que adota uma coloração diferente das demais, visando facilitar a identificação das informações. Por fim, a última coluna contém os ícones correspondentes às ações de edição e exclusão, representados por cores intuitivas que reforçam sua função.

**Figura 27** — Protótipo Modal de Cadastro Funcionário

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

Fonte: Elaborado pelos autores

A Figura 27 apresenta o protótipo da modal de cadastro de funcionários, desenvolvida com o objetivo de tornar o processo de inserção de dados mais direto e funcional. A estrutura da interface segue uma organização em duas colunas. Os campos estão dispostos em uma sequência lógica que facilita o preenchimento por parte do usuário. Na parte inferior da modal, encontram-se os botões “Cancelar” e “Cadastrar”, permitindo, respectivamente, interromper a ação ou concluir o registro. O design mantém a padronização visual das demais telas do sistema, garantindo consistência na experiência do usuário.

Fonte: Elaborado pelos autores

# **REFERÊNCIAS**

BRASIL. **Portaria nº 373, de 2011**, do Ministério do Trabalho e Emprego. Regulamenta os sistemas alternativos de controle de jornada de trabalho, permitindo a adoção de sistemas eletrônicos que não o REP, desde que autorizados em acordo coletivo, e proíbe a marcação automática de ponto, restrições à marcação de ponto e a exigência de autorização prévia para marcação de sobrejornada. Disponível em: <https://www.jusbrasil.com.br/artigos/registro-de-ponto-clt-na-visao-do-ministerio-do-trabalho/1309614245>. Acesso em: 19 fev. 2025.

CERQUEIRA, Izabel. **O que é marcação de ponto e como fazê-la de forma segura?** 2023. Disponível em: <https://solides.com.br/blog/marcacao-de-ponto/>. Acesso em: 25 fev. 2025.

CHIES, Renata. **7 problemas com controle de ponto e como solucioná-los**. 2024. Disponível em: <https://factorialhr.com.br/blog/problemas-controle-de-ponto/>. Acesso em: 26 fev. 2025.

DIMEP - DIMAS DE MELO PIMENTA SISTEMAS DE PONTO E ACESSO. Disponível em: <https://www.dimep.com.br/produtos/relogio-de-ponto-digital/>. Acesso em: 25 fev. 2025.

FILHO, Wilson P. P. **Engenharia de software**: Fundamentos, Métodos e Padrões. 3. ed. Rio de Janeiro: LTC, 2009.

FREEPIK COMPANY S.L. Disponível em: <https://br.freepik.com/fotos-gratis/a-mao-desliza-o-credito-vermelho-atraves-do-terminal-de-pagamento-cartao-de-debito-swiping-on-pos-terminal-maquina-de-leitor-de-cartao-de-credito-no-fundo-branco_1190074.htm#fromView=search&page=1&position=2&uuid=1d90a84e-3bf7-471a-ae31-fb6f21633fa5&query=Registro+De+Ponto+manual>. Acesso em: 25 fev. 2025.

GUEDES, Gilleanes T. A. **UML 2**: Uma abordagem prática. 2. ed. São Paulo: Novatec Editora, 2011.

JUSBRASIL. **Registro de Ponto CLT na visão do Ministério do Trabalho**: Consolidação das Leis Trabalhistas no Brasil. 2022. Disponível em:

<https://www.jusbrasil.com.br/artigos/registro-de-ponto-clt-na-visao-do-ministerio-do-trabalho/1309614245>. Acesso em: 19 fev. 2025.

KL QUARTZ. **Os 7 erros mais comuns na gestão de ponto e como evita-los**. 2024. Disponível em: <https://kl-quartz.com.br/erros-na-gestao-de-ponto>. Acesso em: 21 fev. 2025.

MORANHO, Fernanda. **Ponto Eletrônico Online: porque é a melhor opção?** 2024. Disponível em: <https://tworh.com.br/dp/ponto-eletronico-online-porque-e-a-melhor-opcao/>. Acesso em: 19 fev. 2025.

PRESSMAN, Roger S. **Engenharia de software**: Uma abordagem profissional. 9. ed. Porto Alegre: AMGH, 2021.

REGISTRO PONTO. Disponível em: <https://registroponto.com.br/Solucao>. Acesso em: 25 fev. 2025.

ROGERS, Yvonne; SHARP, Helen; PREECE, Jennifer. **Design de Interação:** Além da interação humano-computador. 3. Ed. Porto Alegre: Bookman Editora, 2013.

SOMMERVILLE, Ian. **Engenharia de Software**. 10. ed. São Paulo: Person Education do Brasil, 2019.

UNGER, Russ; CHANDLER, Carolyn. **O guia para projetar UX**. 1. Ed. Rio de Janeiro: Alta Books, 2009.