***Campus de Cascavel***

**CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS**

**Curso de Bacharelado em Ciência da Computação**

**ANÁLISE ORIENTADA A OBJETOS E IMPLEMENTAÇÃO DE UMA VERSÃO FUNCIONAL DO SISTEMA**

**Sistema de Gerência de Funcionários da Loja “BIG”**

Felipe Kravec Zanatta

Vinicius Vieira Viana

Vitor Mayorca Camargo

CASCAVEL

2024

# LISTA DE FIGURAS

**Figura 1:** Diagrama de Casos de Uso do sistema 7

**Figura 2:** Diagrama de Sequência do Login 14

**Figura 3:** Diagrama de Sequência da visualização do banco de horas próprio 15

**Figura 4:** Diagrama [...] da visualização do banco de horas dos funcionários 16

**Figura 5:** Diagrama de Sequência do cadastro de novos usuários 16

**Figura 6:** Diagrama de Sequência da edição de usuários. 17

**Figura 7:** Diagrama de Classes do Login de Funcionários. 18

**Figura 8:** Diagrama de Classes do Login de Gerentes. 18

**Figura 9:** Diagrama de Classes da Visualização do Próprio Banco de Horas. 18

**Figura 10:** Diagrama de Classes da Visualização do Banco de Horas geral. 19

**Figura 11:** Diagrama de Classes do Cadastro de Usuários. 19

**Figura 12:** Diagrama de Classes da Edição de Usuários. 19

**Figura 13:** Diagrama de Classes das classes << entity >>. 20

**Figura 14:** Diagrama de Classes das classes << entity collection >>. 21

**Figuras 15:** Estrutura das pastas do back-end. 22

**Figuras 16:** Estrutura das pastas do back-end. 22

**Figuras 17:** Estrutura das pastas do front-end. 23

**Figuras 18:** Tela de login final. 24

**Figuras 19:** Menu principal. 25

# 

# SUMÁRIO

[**LISTA DE FIGURAS 1**](#_ut8o6kjvg99q)

[**SUMÁRIO 2**](#_t71dacw9c6e1)

[**1 INTRODUÇÃO 3**](#_nem4imyjywz6)

[1.1 Contexto 3](#_ugh3qan6ee1q)

[1.1.1 Sobre o Processo Atual 3](#_qjv9brni67u5)

[1.2 Metodologia 4](#_9di1hj22mfjg)

[**2 CASOS DE USO ESCOLHIDOS 5**](#_8k3ntx55a06u)

[CASO DE USO: <01> - <Login> 6](#_z8xij45xgha5)

[CASO DE USO: <02> - <Visualização do Próprio Banco de Horas> 8](#_rqcgcly8k6y8)

[CASO DE USO: <03> - <Visualização do Banco de Horas dos Funcionários> 9](#_ob5d1q10shg)

[CASO DE USO: <04> - <Cadastro de Um Novo Usuário> 10](#_28nuoiuifwsf)

[CASO DE USO: <05> - <Editar Um Usuário> 11](#_qdypfgs5kvqo)

[**3 DIAGRAMAS DE SEQUÊNCIA 13**](#_ptbn5gxxslkl)

[CASO DE USO: <01> - <Login> 13](#_danjwcr2lxdb)

[CASO DE USO: <02> - <Visualização do Próprio Banco de Horas> 14](#_d4aq1z2dsf1x)

[CASO DE USO: <03> - <Visualização do Banco de Horas dos Funcionários> 15](#_ty6wrdx9w4lo)

[CASO DE USO: <04> - <Cadastro de Um Novo Usuário> 15](#_cezf75rqhedc)

[CASO DE USO: <05> - <Editar Um Usuário> 16](#_d0s2hfuj8c88)

[**4 DIAGRAMAS DE CLASSES 16**](#_k58shp6pga37)

[CASO DE USO: <01> - <Login> 17](#_90cysytvj6dd)

[CASO DE USO: <02> - <Visualização do Próprio Banco de Horas> 17](#_7h9onibe85ev)

[CASO DE USO: <03> - <Visualização do Banco de Horas dos Funcionários> 18](#_ol1m9j14juui)

[CASO DE USO: <04> - <Cadastro de Um Novo Usuário> 18](#_5ctwz4kkjby2)

[CASO DE USO: <05> - <Editar Um Usuário> 18](#_9few62wy0nlm)

[**5 DIAGRAMA DAS CLASSES ENTITY 19**](#_qdgtipjy8k8t)

[**6 IMPLEMENTAÇÃO 20**](#_q4nu2ndamsft)

[6.1 Back-End: 20](#_v10h7neg7kph)

[6.2 Front-End: 22](#_8uo76h414nby)

[**7 RESULTADOS E CONCLUSÕES 24**](#_wa3vqa9hy3tz)

[**REFERÊNCIAS 26**](#_x4bzke6ioh4y)

[**Apêndice 1 - Sobre a Empresa “BIG Casa” 27**](#_y1y9pek35wvt)

[**Apêndice 2 - ESBOÇO ANTIGO DOS CASOS DE USO 28**](#_65ow5wkeq3hx)

[**Apêndice 3 - Transcrição da Entrevista Sobre o Andamento do Projeto 36**](#_khhzvzg594wh)

[**Apêndice 4 - Requisitos Funcionais 40**](#_ugc716kjjgbq)

[**Apêndice 5 - Requisitos Não Funcionais 45**](#_ut4upi59eb3o)

[**Apêndice 6 - PROTÓTIPOS PRELIMINARES DA INTERFACE 48**](#_hzr3a5tbu637)

# INTRODUÇÃO

## Contexto

A empresa “BIG Casa” (Apêndice 1) atualmente está sob a gestão de Delma Camargo Vieira, e conta com cerca de 18 funcionários. Segundo ela, o processo de gestão desses funcionários ainda é, em muitas partes, feito de maneira manual e arcaica. Um dos processos mais demorados realizados na empresa é o da gestão das horas trabalhadas pelos funcionários, e o subsequente escalonamento dos mesmos para seus dias de folga. Este é um processo manual, e a gerente o executa lentamente com a ajuda de uma caderneta de anotações.

Por conta disso, ela procura por uma forma de automatizar a gerência das horas trabalhadas por seus funcionários, bem como busca uma forma de incentivar os mesmos a “baterem o ponto” nos horários corretos.

### Sobre o Processo Atual

Atualmente, o processo ocorre da seguinte maneira:

* **Registro de Ponto:** Os funcionários preenchem uma folha manualmente: quando chegam eles anotam seu horário de chegada, e quando saem anotam seu horário de saída. Também anotam os horários de saída e chegada do almoço.
* **Escala de folga:** A elaboração da escala de folgas é feita manualmente pelos gerentes, utilizando uma folha de papel com um calendário impresso.
* **Escala de funcionários para datas comemorativas:** Atualmente, os gerentes escalam manualmente quais serão os funcionários que irão fazer extra em datas comemorativas.
* **Acesso ao banco de horas:** Atualmente os funcionários não têm como ter acesso ao banco de horas trabalhadas, já que ele apenas é calculado ao final do mês.
* **Acesso às escalas e folgas:** Atualmente, os funcionários só conseguem obter informações sobre seus dias de folga ou trabalho extra entrando em contato diretamente com o gerente.

## Metodologia

Para entender o problema de Delma e elicitar os requisitos do sistema, a equipe que escreveu o presente documento realizou várias entrevistas com a gerente. Segundo Goguen e Linde (1993), entrevistas são um ótimo método de elicitação de requisitos, pois as informações obtidas durante as mesmas é essencial para que os entrevistadores não façam assumpções errôneas sobre as necessidades de seu entrevistado. As perguntas utilizadas para orientar a entrevista foram baseadas no questionário proposto por Souza e Santander (2011).

Após a entrevista, foi escrito um estudo de viabilidade do sistema, aonde foram explorados diversos métodos para solucionar o problema de Delma. No fim, decidiu-se pela implementação de um website, com foco em mobile, capaz de automatizar todas as tarefas manuais exercidas pelos gerentes da Big Casa. O sistema deve ser capaz de:

* Possibilitar os funcionários a registrar seus horários de saída e entrada;
* Definir automaticamente os dias de folga dos funcionários;
* Informar os funcionários e os gerentes sempre que um dia de folga for definido;
* Escolher quais funcionários irão trabalhar em datas comemorativas;
* Lembrar os funcionários de bater o ponto;

Além disso, os gerentes ou superiores devem ser capazes de ter acesso total às informações armazenadas no sistema referentes a todos os funcionários. Cada funcionário individualmente também deverá conseguir acessar o seu próprio “banco de horas”.

Com base nas decisões feitas no estudo de viabilidade, também foi escrito o documento de especificação de requisitos do sistema. Nele, foi feita a modelagem organizacional dos processos utilizando a metodologia i\* (YU, 1995), a modelagem de processos de negócio utilizando a linguagem gráfica BMPN (CHINOSI E TROMBETTA, 2012), e a especificação dos requisitos funcionais (RF’s) e não-funcionais (RNF’s) do sistema (TURINE; MASIERO, 1996). Os requisitos elicitados para o sistema estão descritos no Apêndices 4 E 5.

Por fim, foram definidos textualmente alguns dos casos de uso do software, uma vez que eles são uma ótima ferramenta para entender o comportamento do sistema, já que nos ajudam a identificar os atores e ações envolvidos (GEPPERT; SCHMID, 2002).

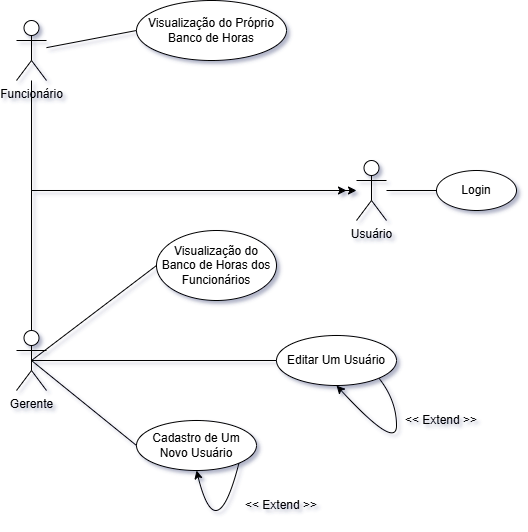
Sendo assim, o presente documento tem como objetivo estudar mais a fundo alguns dos casos de uso do sistema de gerência de funcionários da Big Casa. 5 casos de uso diferentes foram selecionados para estudo.

Estes 5 casos de uso foram utilizados para definir a arquitetura básica do sistema. Para isso, foram feitas: a análise orientada a objetos de cada um deles; a elaboração dos seus diagramas de classes e de sequências; e a elaboração de um diagrama de classes incluindo somente as classes do tipo <<entity>> e <<entity collection>>.

Com a arquitetura básica em mãos, a equipe do projeto iniciou a implementação e completou cerca de 40% do sistema final.

# CASOS DE USO ESCOLHIDOS

O Apêndice 2 mostra um esboço inicial dos casos de uso do projeto, que fora feito para o documento de especificação de requisitos do sistema. 3 desses casos de uso foram reescritos e refinados, e mais 2 casos de uso novos foram escritos para serem estudados no presente documento. Na figura 1 temos o diagrama de casos de uso relacionando eles com os atores do sistema. Nas próximas 7 páginas teremos a descrição textual deles, feita com base numa adaptação do proposto por Cockburn e Cockburn, 2008.

****

**Figura 1:** Diagrama de Casos de Uso do sistema.

#### CASO DE USO: <01> - <Login>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <Permite o Login de Gerentes e Funcionários>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <Gerente e Usuário devem estar cadastrados>

**Condição Final de Sucesso:** <Login bem sucedido>

**Condição Final de Falha:** <Erro no login>

**Ator Primário:** <Usuários e Gerentes>

**CENÁRIO PRINCIPAL DE SUCESSO**

**Passo 1 -** O usuário irá inserir cpf e senha e clicar em “Login”.

**Passo 1.1 -** Caso seja um gerente a fazer login, deverá clicar em “Login gerente”.

**Passo 1.1.2 -** O Gerente irá inserir cpf e senha e clicar em “Login”

**Passo 2 -** As informações serão validadas.

**Passo 3 -** O Usuário receberá a mensagem de “login efetuado com sucesso” e terá acesso às funcionalidades do sistema. Os casos de uso Visualizar banco de horas e bater ponto são incluídos <<include>>

**Passo 3.1 -** O Gerente receberá a mensagem de “login efetuado com sucesso” e terá acesso às funcionalidades do sistema. Os casos de uso Cadastrar novo usuário, Editar usuário, Visualizar banco de horas de todos os funcionários e Visualizar banco de horas de funcionário específico para possível alteração são incluídos <<include>>

**EXTENSÕES**

Passo 2.1, Dados inválidos : Se os dados de login inseridos forem inválidos, o sistema irá enviar a mensagem “Dados incorretos!”, e redirecionará o indivíduo para a tela inicial do sistema. O sistema permitirá apenas 10 tentativas de login a cada hora.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** Extremamente importante, pois garante a privacidade e segurança do sistema>

**Desempenho alvo:** <Pouco menos de um minuto. Tempo suficiente para o usuário inserir as suas informações de login, e esperar o processo de autenticação. Claro que tudo também dependerá da qualidade da conexão com a internet>

**Frequência:** <Será executado toda vez que o usuário acessar o sistema em um novo dispositivo>

**Caso de Uso Pai:** <Caso de uso 4 - Cadastro>

#### CASO DE USO: <02> - <Visualização do Próprio Banco de Horas>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <Permite que o funcionário tenha ciência das suas horas trabalhadas>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <Usuário do tipo “Funcionário” estar autenticado no sistema>

**Condição Final de Sucesso:** <Banco de horas exibido na tela>

**Condição Final de Falha:** <O banco de horas não é exibido na tela>

**Ator Primário:** <Usuário do tipo “Funcionário”>

**CENÁRIO PRINCIPAL DE SUCESSO**

**Passo 1 -** Dentro do menu principal do sistema, o funcionário irá selecionar a aba “Banco de Horas”

**Passo 2 -** O sistema exibe na tela os dados do banco de horas (descrição das horas trabalhadas pelo funcionário) .

**EXTENSÕES**

**Não há.**

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Importante, permite o usuário acompanhar seu registro de horas>

**Desempenho alvo:** <Poucos segundos, o suficiente para abrir o sistema e selecionar a aba do banco de horas>

**Frequência:** <Diferente para cada indivíduo>

#### CASO DE USO: <03> - <Visualização do Banco de Horas dos Funcionários>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <Permite que o gerente tenha uma ideia geral da situação dos seus funcionários>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <Usuário do tipo “Gerente” estar autenticado no sistema>

**Condição Final de Sucesso:** <Banco de horas de todos os funcionários é exibido na tela>

**Condição Final de Falha:** <Os bancos de horas não são exibidos na tela>

**Ator Primário:** <Usuário do tipo “Funcionário”>

---------------------------------------------------------------------------------------------------------------------------

**CENÁRIO PRINCIPAL DE SUCESSO**

**Passo 1 -** Dentro do menu principal do sistema, o gerente irá selecionar a aba “Banco de Horas”

**Passo 2 -** O sistema exibe na tela uma tabela com os dados do banco de horas de cada um dos funcionários.

---------------------------------------------------------------------------------------------------------------------------

**EXTENSÕES**

**Não há.**

---------------------------------------------------------------------------------------------------------------------------

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Importante, facilita a gerência da empresa>

**Desempenho alvo:** <Poucos segundos, o suficiente para abrir o sistema e selecionar a aba do banco de horas>

**Frequência:** <Diária>

#### CASO DE USO: <04> - <Cadastro de Um Novo Usuário>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <Permite a integração de novos usuários ao sistema>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <Usuário do tipo “Gerente” estar autenticado no sistema; Informações básicas do novo usuário>

**Condição Final de Sucesso:** <O usuário estará devidamente cadastrado>

**Condição Final de Falha:** <O usuário não estará cadastrado; O sistema reporta erro e permitirá repetir o processo, reciclando algumas das informações já inseridas>

**Ator Primário:** <Usuário do tipo “Gerente”>

**CENÁRIO PRINCIPAL DE SUCESSO**

**Passo 1 -** Dentro do menu principal do sistema, o gerente selecionará a aba “Cadastrar Novo Usuário”.

**Passo 2 -** O gerente irá inserir no sistema diversas informações básicas sobre o novo usuário, como: Nome de usuário, primeiro nome, sobrenome, CPF, celular, endereço de e-mail, e senha.

**Passo 3 -** O gerente irá apertar o botão “Cadastrar usuário”.

**Passo 4 -** O sistema irá verificar a validade das informações inseridas.

**Passo 5 -** O gerente receberá a mensagem “Usuário cadastrado com sucesso!”, e o novo usuário passará a ter acesso às funcionalidades do sistema.

**EXTENSÕES**

**Passo 5.1:** Dados inválidos: Se algum dos dados inseridos forem inválidos ou estiverem incompletos, o sistema irá enviar a mensagem “Dados inválidos!”, e redirecionará o gerente para a tela de cadastro de novo usuário. O sistema também irá preencher automaticamente alguns dos campos de dados do novo usuário, reciclando as informações inseridas na tentativa que foi invalidada. Isso irá garantir uma maior agilidade no processo de cadastro, e evitará pequenas dores de cabeça << extend >>.

**Passo 5.2:** Dados conflituosos: Se algum dos dados inseridos estiverem em conflito com aqueles de algum usuário já cadastrado, o sistema irá enviar a mensagem “Dados conflitantes com usuário já existente!”. De maneira idêntica ao passo 5.1, o sistema irá redirecionar o gerente à tela de cadastro de usuário, com alguns dos campos já preenchidos, chamando de novo o caso de uso Cadastro de Novo Usuário << extend >>.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** Extremamente importante, pois garante a integração de novos usuários no sistema>

**Desempenho alvo:** <Poucos minutos. Tempo suficiente para o gerente inserir os dados do novo usuário, e esperar o processo de validação. Claro que tudo também dependerá da qualidade da conexão com a internet>

**Frequência:** <Será executado toda vez que um novo funcionário/gerente for integrado ao sistema>

#### CASO DE USO: <05> - <Editar Um Usuário>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <Permite a edição de informações pessoais de usuários ao sistema>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <Usuário do tipo “Gerente” estar autenticado no sistema; Informações do usuário a ser editado>

**Condição Final de Sucesso:** <O usuário será modificado>

**Condição Final de Falha:** <O usuário não será modificado; O sistema reporta erro e permitirá repetir o processo, reciclando algumas das informações já inseridas>

**Ator Primário:** <Usuário do tipo “Gerente”>

**CENÁRIO PRINCIPAL DE SUCESSO**

**Passo 1 -** Dentro do menu principal do sistema, o gerente selecionará a aba “Editar usuário”.

**Passo 2 -** O gerente irá inserir no sistema as novas informações básicas sobre o novo usuário, como: Nome de usuário, primeiro nome, sobrenome, CPF, celular, endereço de e-mail, e senha.

**Passo 3 -** O gerente irá apertar o botão “Atualizar Usuário”.

**Passo 4 -** O sistema irá verificar a validade das informações inseridas.

**Passo 5 -** O gerente receberá a mensagem “Usuário atualizado com sucesso!”, e o novo usuário passará a ter acesso às funcionalidades do sistema.

**EXTENSÕES**

**Passo 5.1:** Dados inválidos: Se algum dos dados inseridos forem inválidos ou estiverem incompletos, o sistema irá enviar a mensagem “Dados inválidos!”, e redirecionará o gerente para a tela de edição. O sistema também irá preencher automaticamente alguns dos campos de dados do novo usuário, reciclando as informações inseridas na tentativa que foi invalidada. Isso irá garantir uma maior agilidade no processo de edição, e evitará pequenas dores de cabeça << extend >>.

**Passo 5.2:** Dados conflituosos: Se algum dos dados inseridos estiverem em conflito com aqueles de algum usuário já cadastrado, o sistema irá enviar a mensagem “Dados conflitantes com usuário já existente!”. De maneira idêntica ao passo 5.1, o sistema irá redirecionar o gerente à tela de edição de usuário, com alguns dos campos já preenchidos, chamando de novo o caso de uso Editar Um Usuário << extend >>.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** Pouco importante, uma vez que um funcionário seja cadastrado de maneira correta, não há necessidade de alteração>

**Desempenho alvo:** <Poucos minutos. Tempo suficiente para o gerente inserir os dados do usuário, e esperar o processo de validação. Claro que tudo também dependerá da qualidade da conexão com a internet>

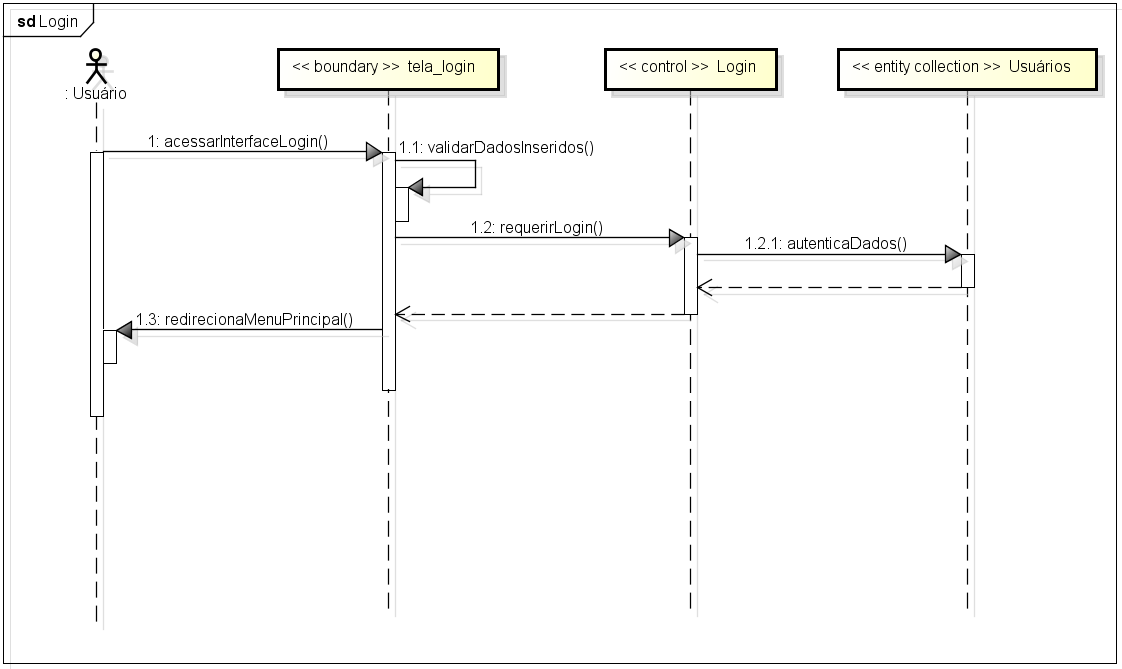
**Frequência:** <Será executado toda vez que um novo funcionário/gerente necessitar de alteração>

# DIAGRAMAS DE SEQUÊNCIA

Diagramas de sequência servem para mostrar o comportamento de casos de uso por meio das interações entre atores e objetos de um sistema, numa sequência cronológica de eventos. Estes diagramas fazem parte do UML.

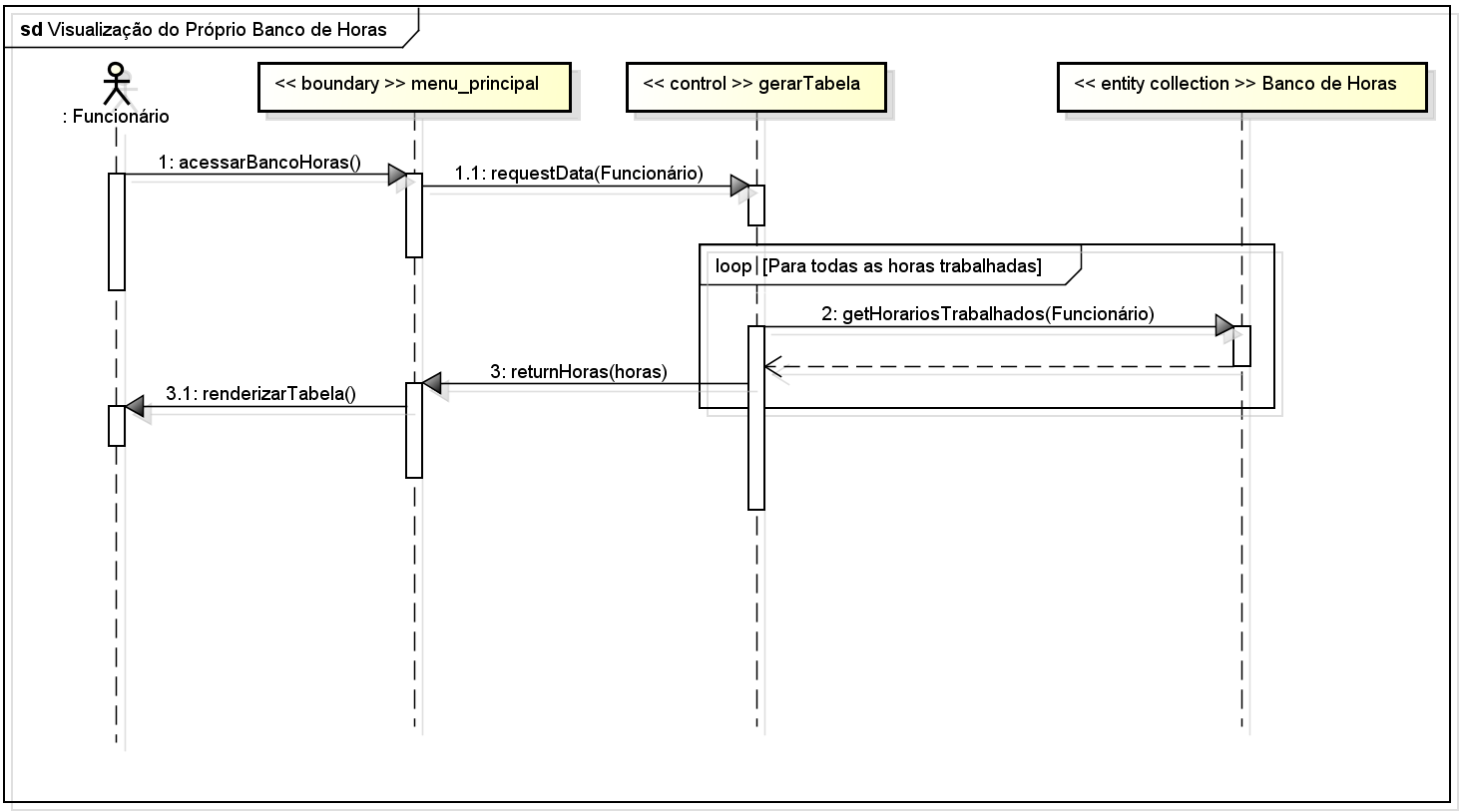
Os diagramas de sequência aqui descritos foram criados seguindo Bennett, McRobb e Farmer (2005). Cada diagrama é composto de no mínimo interfaces <<control>> e <<boundary>>. Control representa funções soltas no sistema, e boundary representa o front-end do sistema. Há um diagrama de sequência para cada caso de uso mencionado acima.

#### CASO DE USO: <01> - <Login>

****

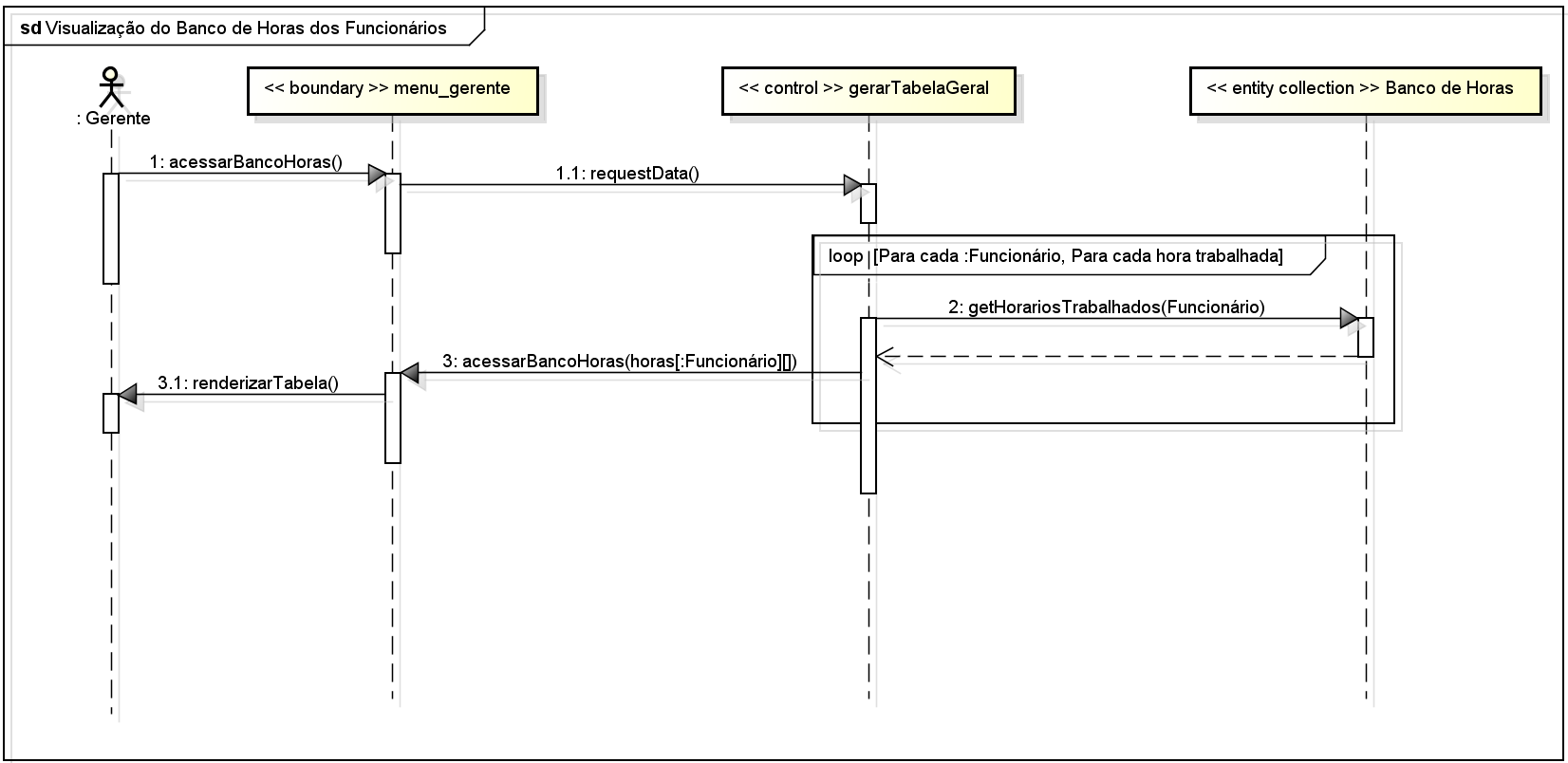
**Figura 2:** Diagrama de Sequência do Login.

#### CASO DE USO: <02> - <Visualização do Próprio Banco de Horas>

****

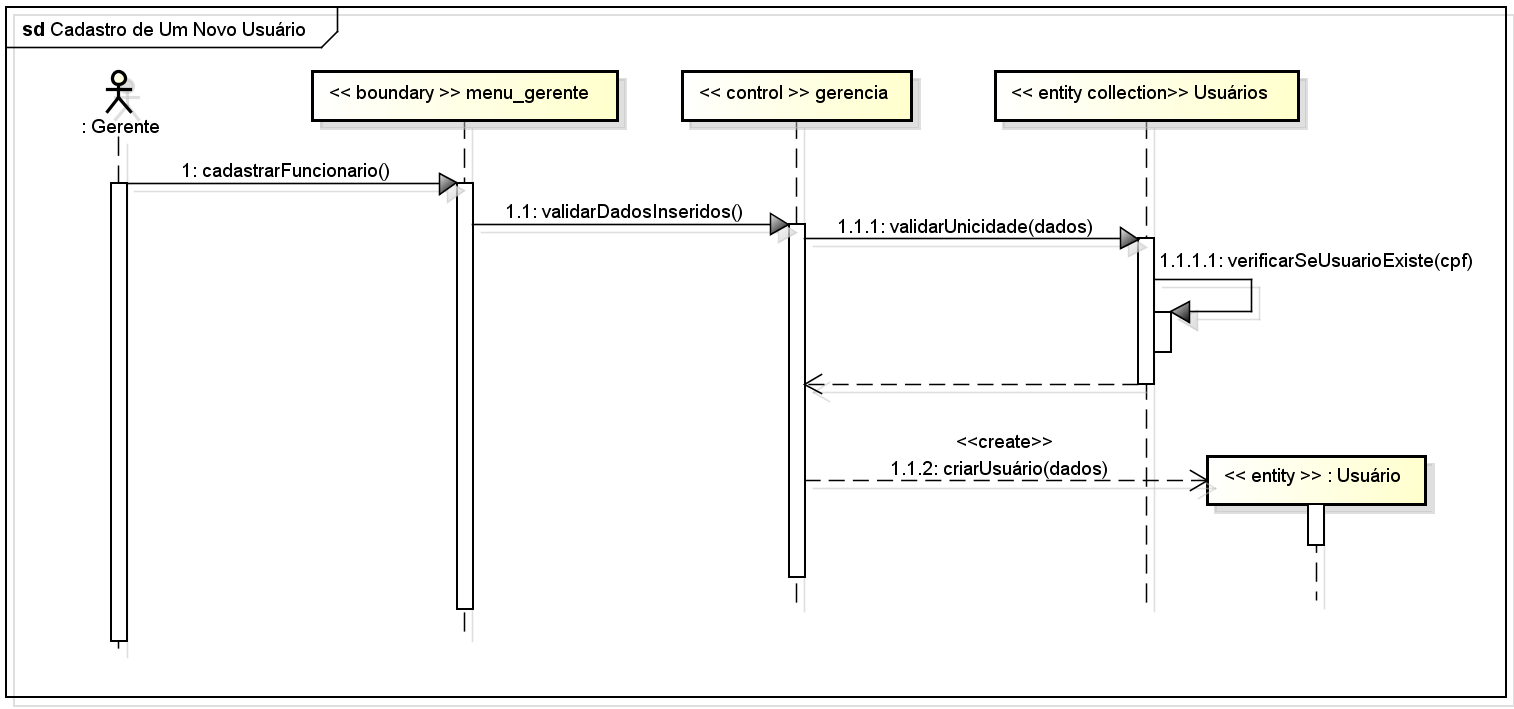
**Figura 3:** Diagrama de Sequência da visualização do banco de horas próprio.

#### CASO DE USO: <03> - <Visualização do Banco de Horas dos Funcionários>

****

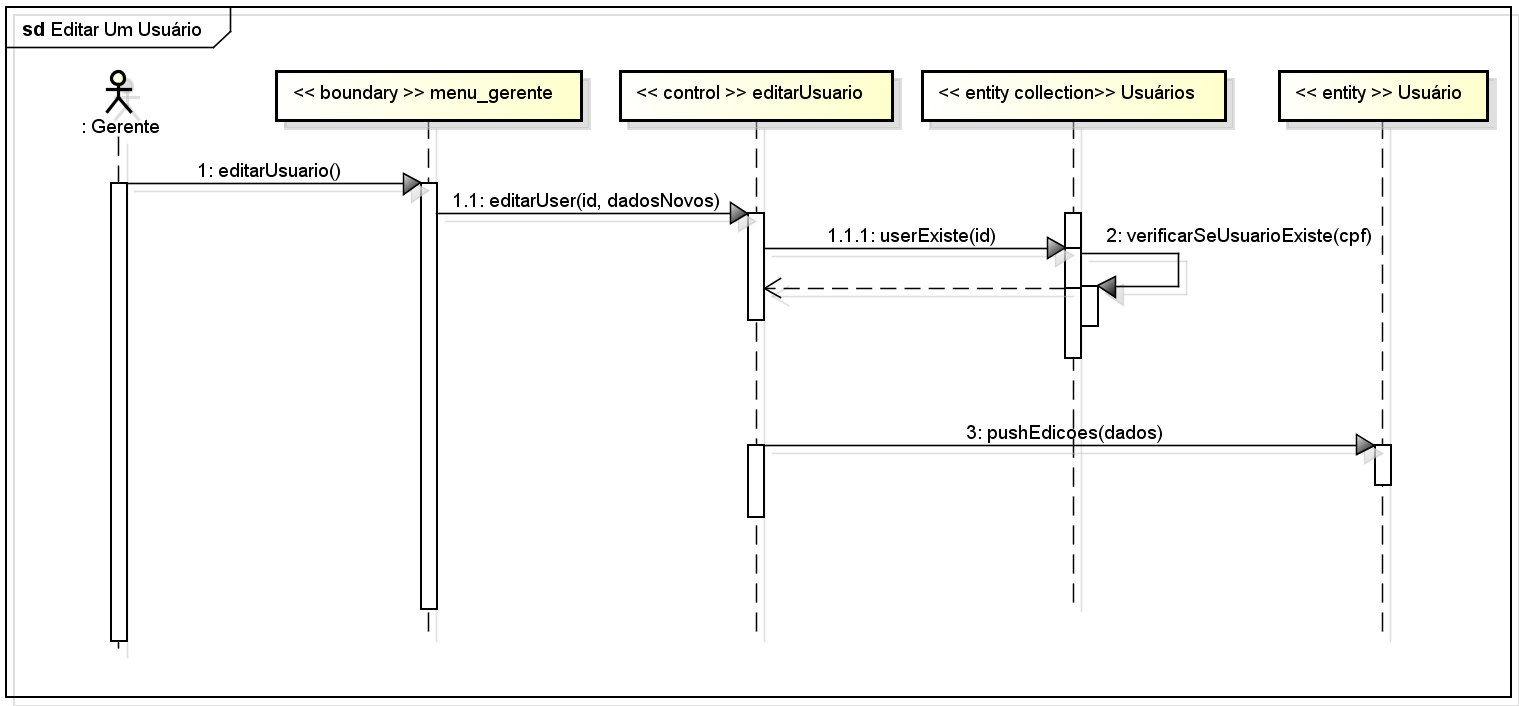
**Figura 4:** Diagrama de Sequência da visualização do banco de horas dos funcionários.

#### CASO DE USO: <04> - <Cadastro de Um Novo Usuário>



**Figura 5:** Diagrama de Sequência do cadastro de novos usuários.

#### CASO DE USO: <05> - <Editar Um Usuário>

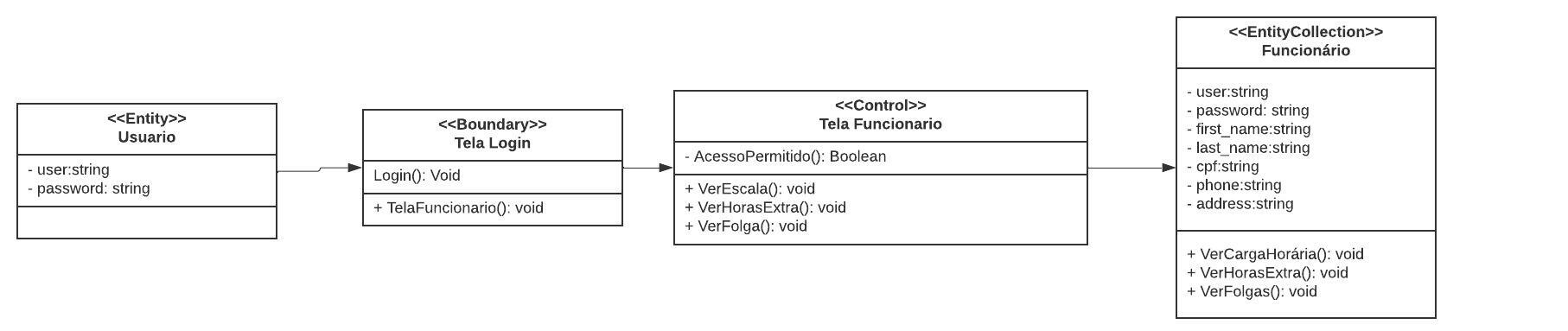


**Figura 6:** Diagrama de Sequência da edição de usuários.

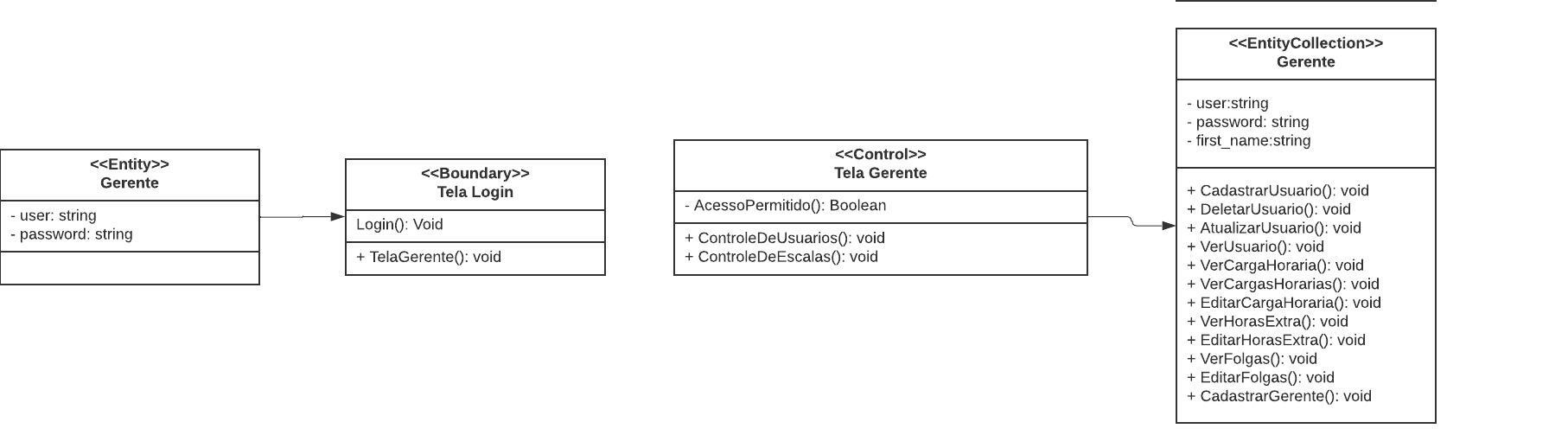
# DIAGRAMAS DE CLASSES

A modelagem dos diagramas de classe foi feita pelo autor responsável pelo Back-End da implementação do sistema. A experiência dele com o framework Django da linguagem Python ajuda a garantir que os planos tenham compromisso com a realidade. Eles estão descritos nas figuras 7, 8, 9, 10, 11 e 12.

#### CASO DE USO: <01> - <Login>



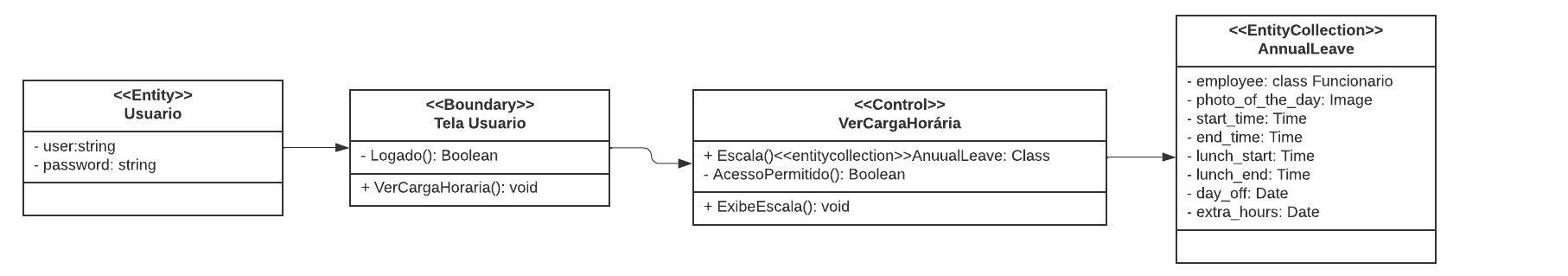
**Figura 7:** Diagrama de Classes do Login de Funcionários.



**Figura 8:** Diagrama de Classes do Login de Gerentes.

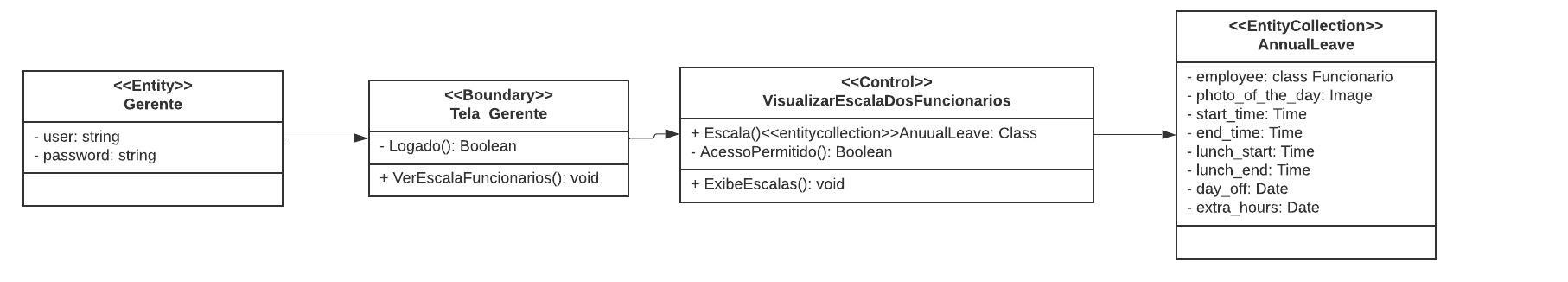
#### 

#### CASO DE USO: <02> - <Visualização do Próprio Banco de Horas>



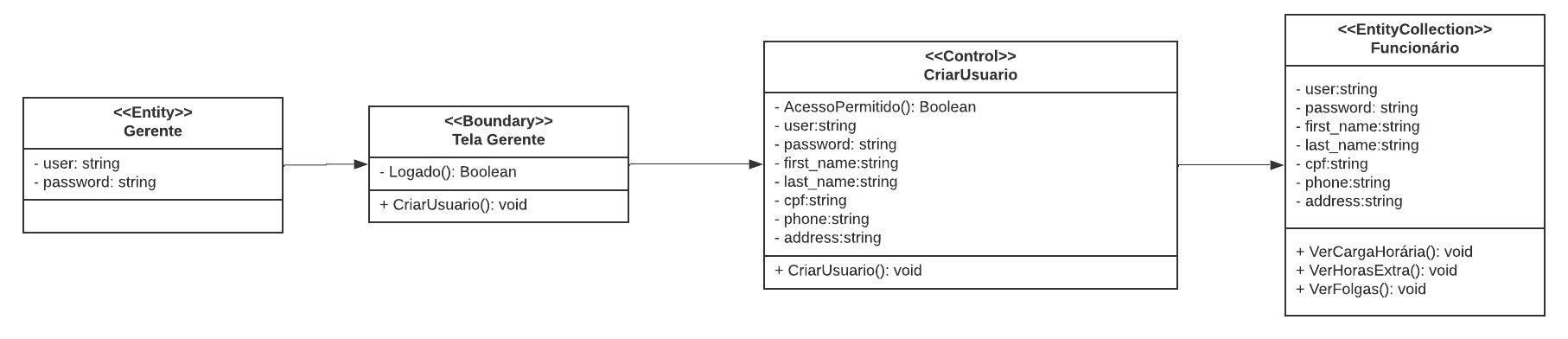
**Figura 9:** Diagrama de Classes da Visualização do Próprio Banco de Horas.

#### CASO DE USO: <03> - <Visualização do Banco de Horas dos Funcionários>



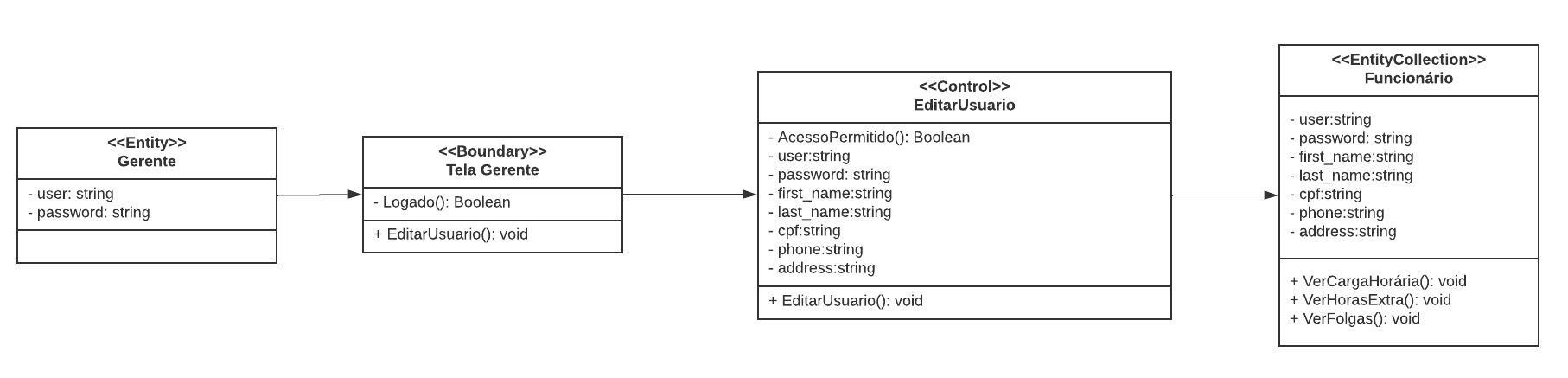
**Figura 10:** Diagrama de Classes da Visualização do Banco de Horas geral.

#### CASO DE USO: <04> - <Cadastro de Um Novo Usuário>



**Figura 11:** Diagrama de Classes do Cadastro de Usuários.

#### CASO DE USO: <05> - <Editar Um Usuário>

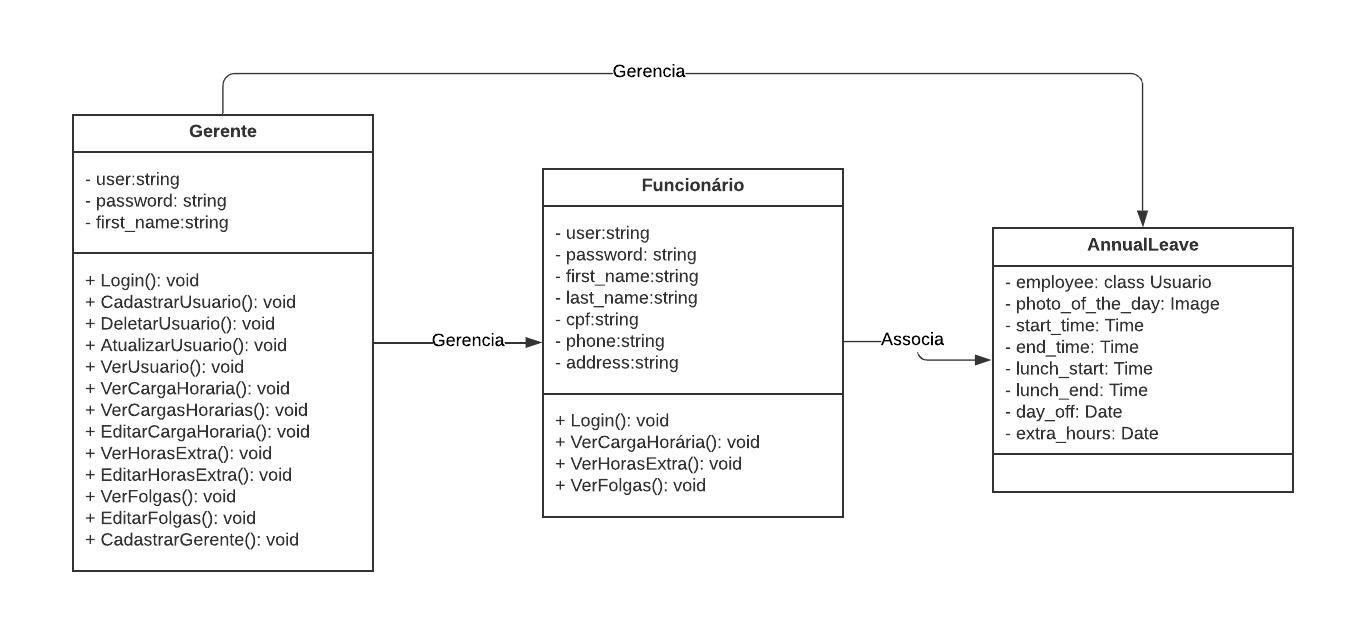


**Figura 12:** Diagrama de Classes da Edição de Usuários.

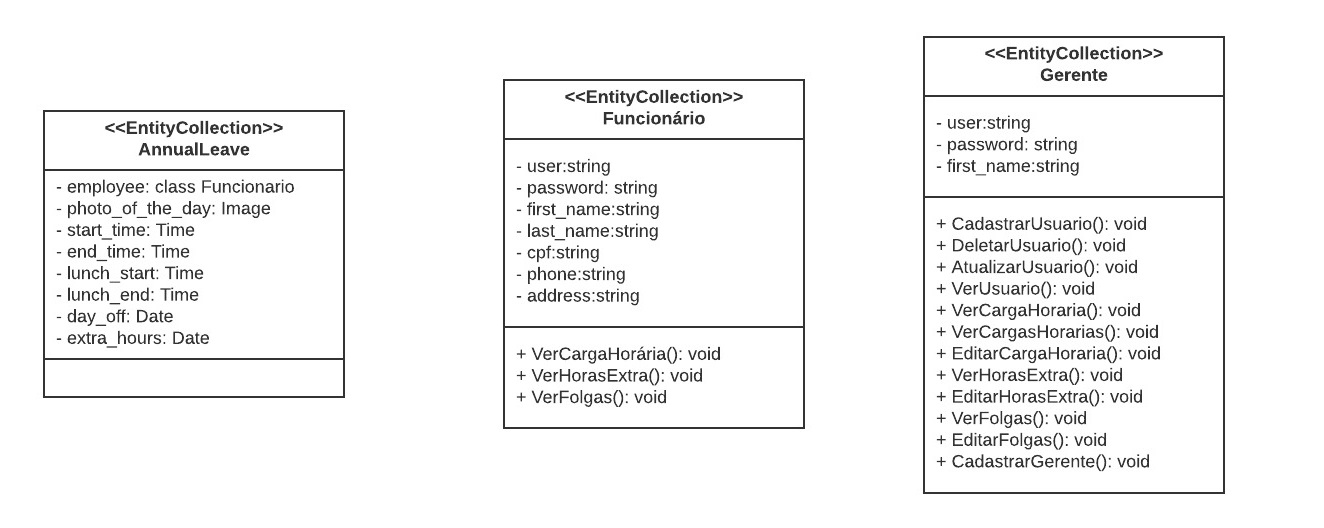
# DIAGRAMA DAS CLASSES ENTITY

Os diagramas das classes de tipo <<entity>> e <<entity collection>>estão descritos nas figuras 13 e 14. Foram implementadas 3 classes principais: Gerente (que gerencia os usuários gerente), Funcionário (que gerencia os usuários funcionário) e AnnualLeave (Classe que é responsável por criar a carga horária de cada Funcionário).

Como Funcionário e AnnualLeave possuem uma relação, sempre que um funcionário é criado no banco de dados, ele é relacionado a uma Table de AnnualLeave do banco de dados. Já a classe Gerente, como tem permissões de super usuário, tem um PGPD (Post, Get, Put e Delete) em cima de ambas as classes, dessa forma, tem a capacidade de gerenciar completamente ambas as Tabelas do banco de dados.

****

**Figura 13:** Diagrama de Classes das classes << entity >>.

****

**Figura 14:** Diagrama de Classes das classes << entity collection >>.

# IMPLEMENTAÇÃO

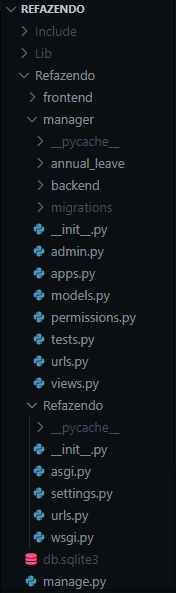
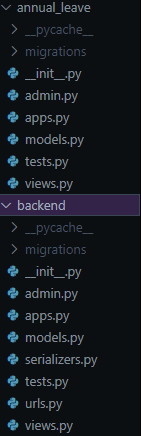
A implementação dos casos de uso selecionados foi dividida em duas partes: front-end e back-end. O back-end foi implementado utilizando a linguagem Python, por meio dos frameworks Django Rest Framework. O front-end foi feito usando as linguagens HTML, CSS e JavaScript, com o uso da biblioteca Bootstrap. Além disso, a interface gráfica foi feita a partir da adaptação dos protótipos de tela (Apêndice 6), previamente aprovados pela cliente.

## Back-End:

Utilizando os frameworks Django e Rest, organizamos nosso projeto em 3 Tables do SQLite: AnnualLeaves (carga horária), Gerente e Usuário (funcionários). Essa alteração foi feita recentemente, já que não tínhamos planos de usar uma table para a carga horária.

Para poder fazer a associação de AnnualLeaves e Usuario e depois gerenciá-los através de Gerente, optou-se por criar sub pacotes dentro de um pacote principal, nesse caso Gerente é o pacote principal e Backend (Usuario) e AnnualLeaves os sub pacotes. Dessa forma, é possível importá-los de um views para outro sem grandes dificuldades.

Organização do backend:

**Figuras 15 e 16:** Estrutura das pastas do back-end.

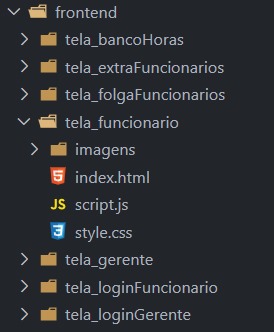
A pasta backend se refere aos arquivos relacionados aos Funcionários. Em geral, foi feito um Post, Get, Put, Delete (PGPD) em cada uma das views, cada uma com sua especificidade e restrição de classes adequada.

Como utilizamos um back-end rest, o front deve ser conectado a ele de forma separada.

## Front-End:

Mostraremos mais à frente a maneira como a equipe implementou o Front-End do projeto. Vale ressaltar que, como este é oprimeiro projeto do grupo utilizando essas tecnologias, não foi utilizado nenhum padrão de criação e organização muito complexo.

O projeto é dividido em telas distintas, cada uma com sua função e objetivo. Foram feitas algumas alterações durante a implementação, devido a necessidades e dificuldades encontradas no caminho. Toda essa parte do Front-End foi feita seguindo os protótipos feitos com o Figma (Apêndice 6).

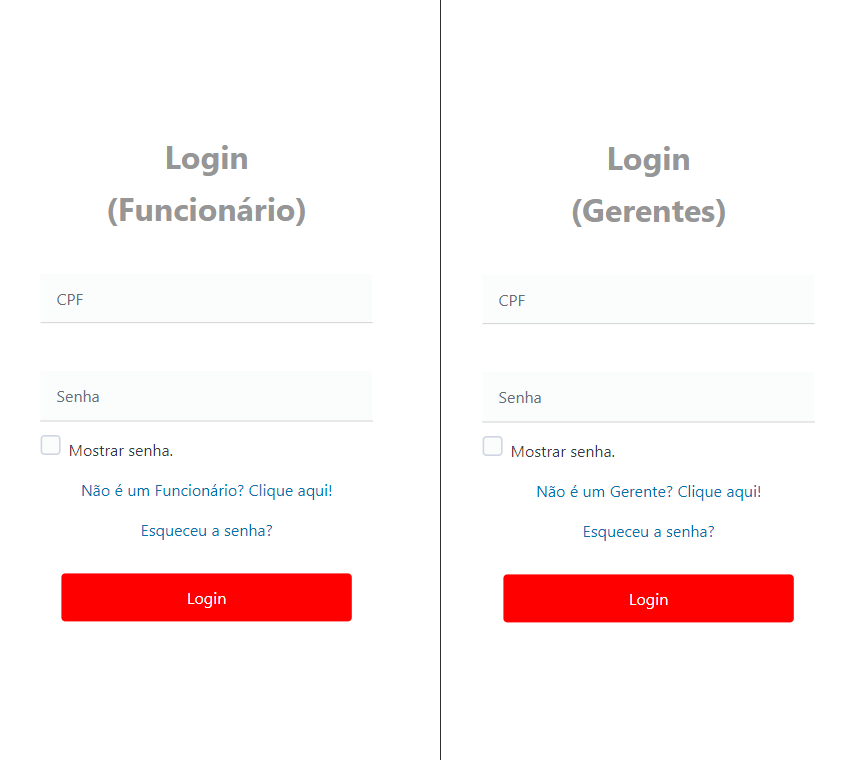


**Figuras 17:** Estrutura das pastas do front-end.

De forma geral, é assim que está organizado nosso front-end, são diversas telas compostas pelas suas marcações HTML5, estilizadas com CSS3, sem a utilização de nenhum framework para facilitar a criação ou até mesmo deixar mais bonito ou organizado.

Poderia ter sido utilizado algum framework, como: Vue JS, React, entre outros. Mas como o grupo não tinha um conhecimento concreto de como fazer a utilização do básico, foi optado que não fosse utilizado nenhum framework.

Podemos visualizar quem temos duas telas de login, uma para funcionários e outra para gerente. Optamos pela criação de duas telas de login diferentes pois ocorreram conflitos técnicos no backend, e a maneira mais fácil de resolver eles é por meio dessa divisão. Ao acessar a tela de login, o usuário irá visualizar um botão perguntando se realmente exerce aquela função. Se não, ele é direcionado para a página correta, conforme a figura 18:



**Figuras 18:** Tela de login final.

Também tivemos alterações na tela de funções, que foi dividida em duas: tela\_gerente e tela\_funcionario. A diferença é uma barra com outras funções além das já mostradas nas telas, essas que apenas o gerente pode acessar clicando em uma barra que está logo acima do usuário logado. Desta forma:



**Figuras 19:** Menu principal.

Fora essas alterações, todo o resto do projeto continua igual ao que foi documentado anteriormente.

# RESULTADOS E CONCLUSÕES

Após a implementação de cerca de 40% do sistema, foi feita mais uma entrevista com Delma Camargo, para entendermos a opinião dela em relação ao andamento do projeto. O arquivo de vídeo da entrevista foi fornecido junto com este documento. Além disso, ela também está transcrita no Apêndice 3.

A cliente demonstrou grande satisfação em relação ao andamento do projeto, e possui grandes expectativas para a finalização do mesmo. Logo, a implementação tem sido um sucesso.

Também vale mencionar que, apesar da equipe ter implementado cerca de 60% do back-end, e 50% do front-end, o processo de “juntar” os dois ainda não foi finalizado. Logo, no momento é impossível colocar no ar uma versão beta do sistema, para que a cliente teste ele e dê o seu feedback.

Com relação aos Requisitos Funcionais (RF’s) e Não-Funcionais (RNF’s) elecitados no documento de requisitos (Apêndices 4 e 5), podemos dizer que o andamento do projeto está em conformidade com eles. O caso de uso 1 (Login) satisfaz os requisitos RF-01 e RF-02; os casos de uso 2 e 3 (Relacionados ao acesso ao Banco de Horas) satisfazem os RF-04 e RF-05; o caso de uso 4 (Cadastro de Um Novo Usuário) satisfaz o RF-03; e, por fim, o caso de uso 5 (Editar Um Usuário) satisfaz o RF-10.

Não houveram muitos requisitos não-funcionais (RNF’s) exigidos pela cliente. Porém, podemos dizer que os RNF-02 (Manutenibilidade), RNF-04 (Integridade), RNF-05 (Segurança), RNF-07 (Baixo Custo), RNF-08 (Confidencialidade) e RNF-10 (Tempo Mínimo de Produção) são garantidos por meio do uso dos frameworks Django e Rest para Python, uma vez que estes são relativamente fáceis de usar, são gratuitos e, principalmente, já tratam da criptografia dos dados de usuários e da integridade do banco de dados embutido.

Além disso, a simplicidade das tecnologias usadas no front-end ajudam na completude dos RNF-02 (Manutenibilidade) e RNF-10 (Tempo Mínimo de Produção). Por fim, vale citar que a entrevista com a cliente indica que o RNF-09 (Interface Simples e Intuitiva), um dos principais RNF’s para ela, está sendo devidamente cumprido, mostrando que a implementação está sendo bem sucedida.

# 

# REFERÊNCIAS

BENNETT, Simon; MCROBB, Steve; FARMER, Ray. **Object-oriented systems analysis and design using UML**. McGraw Hill Higher Education, 2005.

CHINOSI, Michele; TROMBETTA, Alberto. BPMN: An introduction to the standard. **Computer Standards & Interfaces**, v. 34, n. 1, p. 124-134, 2012.

COCKBURN, Alistair; COCKBURN, Lord. **Writing effective use cases**. Pearson Education India, 2008.

GEPPERT, Birgit; SCHMID, Klaus. International Workshop on Requirements Engineering for Product Lines. **ALR**, v. 2002, 2002.

GOGUEN, Joseph A.; LINDE, Charlotte. Techniques for requirements elicitation. **In: [1993] Proceedings of the IEEE International Symposium on Requirements Engineering**. IEEE, 1993. p. 152-164.

SOUZA, Cristiano Ferreira; SANTANDER, Victor Francisco Araya. **Uma Proposta de Elicitação e Análise de Requisitos no Contexto de Médias e Pequenas Empresas de Desenvolvimento de Software**. 14th Workshop On Requirements Engineering. Rio de Janeiro, Brasil, 2011.

TURINE, Marcelo Augusto Santos; MASIERO, Paulo Cesar. **Especificação de requisitos: uma introdução**. 1996.

YU, Eric Siu-Kwong. **MODELLING STRATEGIC RELATIONSHIPS FOR PROCESS REENGINEERING**. 1995. Tese de Doutorado. University of Toronto.

# 

# Apêndice 1 - Sobre a Empresa “BIG Casa”

A BIG Casa é uma marca proeminente no cenário varejista brasileiro, nascida com o propósito de suceder as redes Walmart e Bompreço no país, após a aquisição do Walmart Brasil pela Advent International. Atualmente, a administração da BIG Casa está sob os cuidados do grupo Carrefour Brasil, após a compra do grupo BIG em 2022.

Com uma presença nacional, as lojas da BIG Casa podem ser encontradas em diversas cidades do Brasil. A filial que estamos em contato está localizada próxima à saída da rua Carlos Gomes para a avenida Brasil, inaugurada em 4 de outubro de 2022. Especializada em produtos e artigos para decoração de ambientes, especialmente para residências, a BIG Casa também oferece uma variedade de utilitários, ferramentas, maquiagens, materiais escolares e eletrônicos mais simples.

Dada a sua magnitude e o grande número de colaboradores, a segurança e a integridade das informações dos funcionários são pilares fundamentais para o funcionamento e o crescimento dessa rede. Por isso, a empresa busca constantemente soluções eficazes para o gerenciamento dessas informações.

# 

# Apêndice 2 - ESBOÇO ANTIGO DOS CASOS DE USO

**CASO DE USO**: <01> - <Login>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <O usuário tanto do tipo “funcionário” quanto do tipo “gerente” deseja registrar o login no sistema>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <Os usuários deverão estar conectados à internet. O usuário deve estar dentro dos confins do estabelecimento de trabalho>

**Condição Final de sucesso:** <O usuário fará login no sistema>

**Condição Final de Falha:** <Dados inseridos de forma incorreta>

**Ator Primário:** <Todos os usuários>

**CENÁRIO PRINCIPAL DE SUCESSO**

Passo 1 - Usuário abrirá a aplicação web, e informará seus dados para login.

Passo 2 - O usuário clicará no botão “Login”.

Passo 3 - O sistema exibirá a tela de seleção de opções de acordo com cada tipo de usuário.

**CENÁRIO PRINCIPAL DE FALHA**

Passo 2.1 - Informações incorretas: O usuário digitou informações incorretas e o sistema não foi capaz de achar um usuário e uma senha correspondente.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Alta, sem login, não há operações a serem realizadas>.

**Desempenho alvo:** <Poucos segundos, o suficiente para abrir o sistema, inserir os dados e efetuar o login>.

**Frequência:** Será executada ao menos uma vez ao dia.

**Caso de Uso Pai:** <Não tem>.

**Atores secundários:** <Não possui>.

**CASO DE USO**: <02> - <Controle das Entradas e Saídas dos Funcionários em Horários de Almoço>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <O usuário do tipo “funcionário” deseja registrar os horários em que sai para almoçar e que volta do almoço>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <O usuário deverá estar previamente logado no sistema; O usuário deve estar dentro dos confins do estabelecimento de trabalho>

**Condição Final de sucesso:** <Horários de entrada/saída são salvos no banco de dados>

**Condição Final de Falha:** <Horários de entrada/saída não são salvos no banco de dados; O sistema reporta erro e pede para repetir o processo>

**Ator Primário:** <Usuário do tipo “funcionario>

**CENÁRIO PRINCIPAL DE SUCESSO**

Passo 1 - O usuário abrirá a aplicação web e selecionará a aba de registros de entrada e saída <<include>>.

Passo 2 - O usuário escolherá a opção “Registrar entrada/saída”.

Passo 3 - O sistema irá pedir para o usuário tirar uma foto - dentro dos confins do estabelecimento – para provar que ele estava no local de trabalho no momento.

Passo 4 - O sistema registrará o horário de entrada/saída do funcionário e salvará tais dados no banco de horas.

**CENÁRIO PRINCIPAL DE FALHA**

Passo 2.1 - Erro de conexão: O registro do banco de horas depende da conexão com o banco de dados da empresa. Se não houver conexão, os dados ficarão salvos temporariamente no dispositivo do usuário.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Extremamente importante, pois permite fazer com maior precisão o cálculo do banco de horas de cada funcionário>.

**Desempenho alvo:** <Poucos segundos, o suficiente para abrir o sistema, selecionar a área de registros de entrada/saída, e apertar o botão. Claro que tudo também dependerá da qualidade da conexão com a internet>.

**Frequência:** Será executado duas vezes ao dia.

**Caso de Uso Pai:** <Caso de uso 01 - Login>.

**Atores secundários:** <Usuários do tipo gerente>.

**CASO DE USO**: <03> - <Inserir/Alterar dados pessoais>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <O usuário do tipo “funcionário” deseja inserir suas informações pessoais no sistema>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <O usuário deverá estar previamente logado no sistema; O usuário deve estar dentro dos confins do estabelecimento de trabalho>

**Condição Final de sucesso:** <Informações salvas no banco de dados>

**Condição Final de Falha:** <O usuário inseriu informações inválidas>

**Ator Primário:** <Usuário do tipo “funcionario>

**CENÁRIO PRINCIPAL DE SUCESSO**

Passo 1 - Usuário abrirá a aplicação web e selecionará a aba de informações pessoais<<include>>.

Passo 2 - O usuário selecionará a opção que melhor lhe convém.

Passo 3 - O sistema mostrará todas as opções cadastradas até o momento, as que não estiverem cadastradas estarão em branco.

Passo 4 - O usuário editará ou inserirá as informações nos campos correspondentes.

Passo 5 - O sistema salvará no banco de dados as informações do usuário.

**CENÁRIO PRINCIPAL DE FALHA**

Passo 5.1 - O usuário inseriu informações como CPF ou RG incorretas. O sistema levantará uma exceção e não será possível atualizar ou cadastrar as informações caso o erro não seja corrigido.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Extremamente importante, pois permite com que os gerentes busquem informações dos usuários de forma rápida e prática>.

**Desempenho alvo:** <Alguns minutos, é necessário cadastrar informação a informação, tornando um processo lento e manual>.

**Frequência:** Será executado raramente.

**Caso de Uso Pai:** <Caso de uso 01 - Login>.

**Atores secundários:** <Usuários do tipo gerente>.

**CASO DE USO**: <04> - <Usuário deseja verificar a própria escala>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <O usuário do tipo “funcionário” deseja visualizar sua escala>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <O usuário deverá estar previamente logado no sistema.>

**Condição Final de sucesso:** <O usuário visualizou e verificou sua escala>

**Condição Final de Falha:** <Erro de comunicação com o banco de dados.>

**Ator Primário:** <Usuário do tipo “funcionario>

**CENÁRIO PRINCIPAL DE SUCESSO**

Passo 1 - O usuário abrirá a aplicação web e selecionará a aba de “Visualização da escala”.

Passo 2 - O usuário visualiza sua escala.

**CENÁRIO PRINCIPAL DE FALHA**

Passo 2.1 - Erro de comunicação com o banco de dados: O registro do banco não foi capaz de recuperar as informações relacionadas à escala do funcionário.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Importante, permite com que o funcionário saiba quais dias trabalha e quais folga>.

**Desempenho alvo:** <Poucos segundos, o suficiente para abrir o sistema, selecionar a aba da escala, e visualizá-la.>.

**Frequência:** Diferente para cada funcionário individual.

**Caso de Uso Pai:** <Caso de uso 01 - Login>.

**Atores secundários:** <Não possui>.

**CASO DE USO**: <05> - <Funcionário deseja verificar as horas pendentes>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <O usuário do tipo “funcionário” deseja verificar quantas horas precisa para fechar a escala mensal>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <O usuário deverá estar previamente logado no sistema.>

**Condição Final de sucesso:** <O funcionário visualiza as horas pendentes>

**Condição Final de Falha:** <<Erro de comunicação com o banco de dados.>

**Ator Primário:** <Usuário do tipo “funcionario>

**CENÁRIO PRINCIPAL DE SUCESSO**

Passo 1 - O usuário abrirá a aplicação web e selecionará a aba para visualizar a escala.

Passo 2 - No topo da tela, serão exibidas as horas pendentes do usuário.

**CENÁRIO PRINCIPAL DE FALHA**

Passo 2.1 - Erro de comunicação com o banco de dados: O registro do banco não foi capaz de recuperar as informações relacionadas ao banco de horas do funcionário.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Média importância, permite ao usuário um controle um pouco melhor de quanto falta para bater sua meta>.

**Desempenho alvo:** <Poucos segundos, o suficiente para abrir o sistema, selecionar a área da escala.

**Frequência:** Diferente para cada usuário individual.

**Caso de Uso Pai:** < Caso de uso 04 - Usuário deseja verificar a própria escala>.

**Atores secundários:** <Usuários do tipo gerente>.

**CASO DE USO**: <06> - <O funcionário deseja visualizar o banco de horas>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <O usuário do tipo “funcionário” deseja verificar seus dias e horas já cumpridas>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <O usuário deverá estar previamente logado no sistema.>

**Condição Final de sucesso:** <O usuário visualiza seu banco de horas>

**Condição Final de Falha:** <Erro de comunicação com o banco de dados>

**Ator Primário:** <Usuário do tipo “funcionario>

**CENÁRIO PRINCIPAL DE SUCESSO**

Passo 1 - O usuário abrirá a aplicação web e selecionará a aba de escala.

Passo 2 - O usuário escolherá a opção “Visualizar Banco de horas”.

Passo 3 - O sistema exibirá o banco de dados ao funcionário

**CENÁRIO PRINCIPAL DE FALHA**

Passo 3.1 - Erro de comunicação com o banco de dados: O registro do banco não foi capaz de recuperar as informações relacionadas ao banco de horas do funcionário.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Pouco importante, permite o usuário acompanhar seu registro de horas>.

**Desempenho alvo:** <Poucos segundos, o suficiente para abrir o sistema, selecionar a aba da escala e do banco de horas>.

**Frequência:** Diferente para cada usuário individual.

**Caso de Uso Pai:** <Caso de uso 04 - Usuário deseja verificar a própria escala>.

**Atores secundários:** <Usuários do tipo gerente>.

**CASO DE USO**: <07> - <Solicitação de mudança na escala do usuário>

**INFORMAÇÃO CARACTERÍSTICA:**

**Objetivo no Contexto:** <O usuário do tipo “funcionário” solicita uma alteração em sua escala>

**Escopo:** <Sistema de Gerência de Funcionários da Loja “BIG”>

**Pré-condições:** <O usuário deverá estar previamente logado no sistema.>

**Condição Final de sucesso:** <O gerente é notificado da solicitação>

**Condição Final de Falha:** <O gerente não é notificado>

**Ator Primário:** <Usuário do tipo “funcionario>

**CENÁRIO PRINCIPAL DE SUCESSO**

Passo 1 - O usuário abrirá a aplicação web e selecionará a aba de escala.

Passo 2 - O usuário escolherá a opção “Solicitar alteração na escala”.

Passo 3 - O usuário insere a data que deseja a alteração da escala.

Passo 4 - O sistema enviará uma notificação ao gerente para a mudança na escala.

Passo 5 - A escala é alterada <<include>>.

**CENÁRIO PRINCIPAL DE FALHA**

Passo 2.1 - Erro de comunicação com o servidor: O gerente não foi notificado do pedido de alteração da escala.

**INFORMAÇÃO RELACIONADA**

**Prioridade:** <Média importância, permite que o gerente realize mudanças rápidas nas escalas de seus funcionários>.

**Desempenho alvo:** <Alguns minutos, talvez horas, é necessário que o gerente seja notificado e tenha tempo hábil para realizar as trocas>.

**Frequência:** Individual para cada funcionário.

**Caso de Uso Pai:** <Caso de uso 01 - Login>.

**Atores secundários:** <Usuários do tipo gerente>.

# Apêndice 3 - Transcrição da Entrevista Sobre o Andamento do Projeto

- Vinicius: Essa aqui é uma entrevista com a Delma sobre o trabalho de Paas.

- Vinicius: A gente vai fazer umas perguntas sobre o que ela tá achando até agora do visual, do front-end do projeto.

- Vinicius: Essa tela inicial aqui é um login para funcionários.

- Vinicius: AA gente pode vir aqui.

- Vinicius: A usuária vai digitar o CPF dele.

- Vinicius: CPF qualquer, vamos colocar qualquer senha aqui por enquanto.

- Vinicius: Login.

- Vinicius: Bom, ele vai fazer o login aqui, ignora essa fotinha feia.

- Vinicius: Bom, isso aqui é o login para funcionário.

- Delma: Certo.

- Vinicius: Aí, caso você é um gerente, você vai clicar aqui, ele troca para a tela de gerente e funciona da mesma forma.

- Vinicius: Você vai fazer o login aqui.

- Vinicius: E bom, vai ser mandado para cá.

- Vinicius: Os dois vão te trazer para o mesmo lugar.

- Vinicius: A única mudança vai ser que o gerente vai poder fazer isso aqui.

- Vinicius: Jogar essa janelinha para cima, enquanto o funcionário não vai poder.

Dessa parte de login, o que você está achando?

- Delma: Muito interessante.

- Delma: Nesse caso, isso vai ser armazenado para mim poder verificar, por exemplo,

se o meu funcionário bateu ponto ou não bateu.

- Vinicius: Não, aqui ainda não é para bater ponto.

- Vinicius: É só para entrar no aplicativo.

- Delma: Muito bom, tá.

- Vinicius: Ok.

- Vinicius: Aí a gente tem aqui as opções que o funcionário vai poder ter acesso, que

é o banco de horas.

- Vinicius: Aqui ele vai poder olhar o banco de horas dele.

- Delma: Que ele tem pra tirar ou tá bebendo.

- Vinicius: Quer dizer, perdão, esse aqui é do dia a dia e no total também, mas

principalmente do dia a dia.

- Vinicius: Tipo, a hora que ele entra, sai pro almoço Entra de tarde e sai de novo.

- Vinicius: Aí o total de horas que ele fez no dia.

- Delma: Certo, isso é ótimo.

- Vinicius: Isso vai ficando nas linhas assim, e lá no total, lá no final, uma linha pra

baixo aqui, vai ficar o total que ele fez no mês.

- Vinicius: ok

- Vinicius: Uma das coisas que tinha que ter.

- Delma: Por exemplo, no caso, se não me falha a memória, São 220 horas-mes que

um colaborador num horário de trabalho regular seria, né?

- Delma: Eu acredito que sim.

- Delma: Nessa soma, então, a gente teria ou um déficit ou uma sobra dentro de

hora.

- Delma: Então a gente conseguiria saber se esse funcionário tem hora pra tirar ou

Pra pagar.

- Vinicius: Exatamente.

- Vinicius: Aí a gente tem essa parte de fogo dos funcionários, onde vai mostrar Qual

funcionário tem folga em tal dia?

- Vinicius: Qual não tem?

- Vinicius: Qual vai ter que fazer?

- Vinicius: Qual que não vai ter?

- Vinicius: E assim vai.

- Vinicius: É uma página só pra mostrar uma informação.

- Vinicius: Aqui tem uma página basicamente igual, só que pra mostrar qual que vai

Fazer extra e qual que não vai fazer extra.

- Vinicius: Vai tá tipo falando, ah, funcionário tal tem que fazer extra tal dia.

- Delma: Pra poder pagar a hora que tá devendo, você quer dizer?

- Vinicius: É, e também é...

- Vinicius: negócio dos..

- Vinicius: como é que é?

- Vinicius: Nantes feriados, é..

- Vinicius: datas comemorativas, né, que geralmente faz aquele giro dos funcionários

lá.

- Delma: Sim, sim, aham.

- Delma: E é metade dos funcionários só que vai fazer a hora deles, né?

- Vinicius: Aham, exatamente.

- Vinicius: É.

- Vinicius: ok.

- Vinicius: Aí, como gerente, você tem umas opções a mais.

- Vinicius: Vindo aqui.

- Vinicius: você vai ter essas três opções aqui, que é adicionar um funcionário, você

preenche esses campos aqui, quando você termina de preencher, por

exemplo, Vinícius, Vieira, Viana, que é eu, o número de telefone, qualquer

número, CPF, qualquer coisa aqui, o e-mail, aí ele vai dar, ele vai Ah, ele

confirma que tem um e-mail válido aqui, cacete.

- Vinicius: Por isso aqui, automático?

- Vinicius: Não, não dá.

- Vinicius: Vamos colocar um aleatório aqui..

- Vinicius: AÍ após...

- Vinicius: Então, coloca um e-mail válido, ele libera o botão...

- Vinicius: Ah, faz o corte depois.

- Vinicius: Aí ele libera o botão e a gente consegue fazer o cadastro do funcionário

novo.

- Vinicius: Aí aqui a gente consegue editar um funcionário já existente.

- Vinicius: Não existe ainda, por enquanto.

- Vinicius: Não era uma das nossas prioridades na nossa construção atualmente.

- Vinicius: E nem eu excluí funcionário.

- Vinicius: Mas bom, é autodidático o que cada uma dessas coisas vai poder fazer.

- Vinicius: Vou editar funcionário, você vai pegar um funcionário...

- Delma: É interessante, excluir quando é necessário e editar quando é necessário.

- Vinicius: Exatamente.

- Vinicius: Basicamente é isso o que você vai poder fazer como gerente.

- Vinicius: A partir do editar funcionário você pode também personalizar quando ele

vai fazer uma hora extra ou quando ele vai folgar, caso necessário.

- Vinicius: Aqui onde está esse textinho do usuário logado vai estar.

Transcrição\_entrevista.md

Exibindo Transcrição\_entrevista.md…

# Apêndice 4 - Requisitos Funcionais

As abreviações utilizadas são: ID (Identificação), RF (Requisito Funcional), DESC (Descrição), RAT (Racionalidade, motivo), DEP (Dependências), e PRI (Prioridade).

**ID:** [RF-01]

**TÍTULO:** Sistema de Login

**DESC:** Para que o usuário possa acessar as funcionalidades do sistema, ele deve primeiro passar por um processo chamado de “login”. Durante o login, o usuário deverá informar as suas informações de acesso ao sistema (“nome do usuário” e “senha”). O usuário terá acesso às funcionalidades do sistema se e somente se os dados inseridos forem condizentes com aqueles salvos no banco de dados do aplicativo.

**RAT:** Necessário para evitar fraudes e garantir a segurança e integridade do sistema.

**DEP:** Nenhuma.

**PRI:** Alta.

**ID:** [RF-02]

**TÍTULO:** Duas Classes de Usuário com Diferentes Níveis de Acesso ao Sistema

**DESC:** Os usuários cadastrados no sistema estarão divididos em duas categorias diferentes: “Funcionários” e “Gerentes”. Estes terão acesso a diferentes funcionalidades do software.

Ao realizar o login, funcionários só poderão acessar funcionalidades como o próprio banco de horas, seu histórico de horas trabalhadas, momentos em que bateu o ponto e a sua escala de folgas.

Por outro lado, os gerentes serão capazes de acessar todas as informações referentes a todos os funcionários da empresa, bem como serão capazes de criar, alterar e deletar os funcionários cadastrados no sistema.

**RAT:** Necessário para que funcionários comuns não tenham acesso a certas funções do sistema.

**DEP:** [RF-01].

**PRI:** Alta.

**ID:** [RF-03]

**TÍTULO:** Cadastro de funcionários por dentro do sistema

**DESC:** Usuários do tipo “gerente” deverão ser capazes de cadastrar novos usuários de qualquer tipo no sistema. Ao cadastrar um usuário novo, o sistema precisará de algumas informações básicas como: “nome de usuário”, “senha”, “tipo de usuário” e quaisquer outras informações básicas que sejam necessárias para identificar o usuário dentro do sistema.

**RAT:** Necessário para que os gerentes possam integrar seus funcionários no sistema quando quiserem.

**DEP:** [RF-01], [RF-02].

**PRI:** Média.

**ID:** [RF-04]

**TÍTULO:** Banco de Horas.

**DESC:** Anexo ao aplicativo haverá um banco de dados que irá armazenar as horas trabalhadas de cada um dos funcionários, bem como os momentos de descanso, as horas que o funcionário entrou e saiu da empresa, e os momentos em que ele bateu o ponto. Posteriormente, estas informações serão processadas pelo sistema.

**RAT:** Todo o sistema irá se basear em volta dos dados armazenados no banco de horas.

**DEP:** Nenhuma.

**PRI:** Altíssima.

**ID:** [RF-05]

**TÍTULO:** Acesso ao Banco de Horas

**DESC:** Cada usuário deverá ser capaz de acessar as informações contidas no banco de horas do sistema [RF-03], dependendo do seu tipo de usuário.

Um usuário do tipo “funcionário” só será capaz de acessar informações referentes a ele mesmo, como por exemplo: total de horas trabalhadas, momentos em que entrou, saiu da empresa e bateu o ponto e quantidade de horas extras trabalhadas.

Usuários do tipo “gerente” poderão acessar todas as informações referentes a todos os funcionários.

**RAT:** Essencial para que usuários consultem as horas pagas e pendentes deles ou de outros usuários, e para a gestão da empresa por parte dos gerentes.

**DEP:**. [RF-04].

**PRI:** Altíssima.

**ID:** [RF-06]

**TÍTULO:** Registro de Entrada e Saída dos Funcionários (“Sistema de ponto”).

**DESC:** Os funcionários irão poder registrar seus horários de saída e entrada do expediente, inclusive horário de saída para almoço e o retorno, essas horas serão registradas no banco de horas do [RF-04].

Isso será feito através de uma aba no aplicativo, aonde será registrado o horário e uma foto do funcionário dentro da área da loja.

**RAT:** Essencial para determinar o salário e os dias de descanso dos funcionários.

**DEP:** [RF-04].

**PRI:** Alta.

**ID:** [RF-07]

**TÍTULO:** Escalonamento da Folga dos Funcionários.

**DESC:** O app deverá ser capaz de usar as informações armazenadas no [RF-04] para definir automaticamente os dias de folga de cada funcionário.

A loja é dividida em vários setores distintos e, devido à sua quantidade de funcionários, pode folgar apenas um funcionário por dia por setor. O sistema não poderá fazer com que 2 (dois) funcionários de um mesmo setor tenham folga no mesmo dia, e deve levar em consideração que cada funcionário tem direito a um dia de folga a cada 3 sábados trabalhados. O dia de folga de funcionário (segunda, terça, quarta...) não pode repetir, isso para possibilitar que todos os funcionários tenham a possibilidade de folgar em todos os dias da semana.

**RAT:** Essencial para determinar os dias de folga dos funcionários.

**DEP:** [RF-04], [RF-06].

**PRI:** Altíssima.

**ID:** [RF-08]

**TÍTULO:** Escalonamento de Funcionários para Datas Comemorativas.

**DESC:** De forma similar ao [RF-07], o aplicativo também deverá ser capaz de gerenciar quais funcionários deverão trabalhar, e quais funcionários receberão folga em datas comemorativas (Natal, Páscoa, Ano-Novo).

O escalonamento funcionará de forma que metade de todos os funcionários deverão fazer hora extra até mais tarde, e deverá haver um revezamento desses funcionários. Detalhes específicos dessa parte do sistema deverão ser posteriormente definidos com a gerente da loja.

**RAT:** Essencial para gerenciar o funcionamento da loja em datas comemorativas.

**DEP:** [RF-04], [RF-06].

**PRI:** Alta.

**ID:** [RF-09]

**TÍTULO:** Alteração manual das datas de folga dos funcionários.

**DESC:** Todo usuário do tipo “Gerente” deverá ser capaz de manualmente alterar os dias em que cada funcionário receberá folga. O Gerente poderá conceder folgas manualmente conforme achar necessário, como por exemplo, mas não limitado a: para funcionários com uma grande quantidade de horas extras, funcionários que tiverem trabalhado em muitos sábados e datas comemorativas, ou funcionários que tiverem negociado suas folgas previamente com algum gerente.

**RAT:** Necessário facilitar a gestão dos dias de folga dos funcionários da empresa.

**DEP:** [RF-02],[RF-04], [RF-07], [RF-08].

**PRI:** Alta.

**ID:** [RF-10]

**TÍTULO:** Alteração manual dos dados pessoais dos funcionários.

**DESC:** Todo usuário do tipo “Gerente” deverá ser capaz de manualmente alterar os dados e informações pessoais de qualquer “Funcionário cadastrado”. Isso inclui, por exemplo, a capacidade de alterar a senha de algum Funcionário que a tenha esquecido.

**RAT:** Necessário facilitar a gestão dos funcionários da empresa.

**DEP:** [RF-02].

**PRI:** Média.

**ID:** [RF-11]

**TÍTULO:** Lembretes quanto ao número de horas trabalhadas e à escala de folgas.

**DESC:** A cada 15 dias, o aplicativo deverá enviar um lembrete a todos os funcionários, informando o número de horas restantes para serem pagas, o número de horas extras trabalhadas, o número de dias de trabalho que faltam para receber uma folga, e os dias de folga ao qual lhe foram concedidos.

Da mesma forma, a cada 15 dias os gerentes serão informado de quais funcionários receberam folga, e quando. Da mesma forma, o gerente será informado de quais funcionários estão devendo muitas horas de trabalho, e quais que já possuem um número muito alto de horas extras trabalhadas.

**RAT:** Necessário para que os funcionários e gerentes estejam cientes de suas horas trabalhadas.

**DEP:** [RF-04], [RF-07], [RF-08].

**PRI:** Altíssima.

**ID:** [RF-12]

**TÍTULO:** Checar Quais Funcionários Estão Trabalhando no Momento.

**DESC:** Usuários do tipo “Gerente” deverão ter acesso à uma função especial onde poderão checar quais funcionários já bateram ponto, e quais estão presentes no estabelecimento de trabalho no momento atual.

**RAT:** Útil para facilitar a gestão dos funcionários.

**DEP:** [RF-02], [RF-06].

**PRI:** Média.

**ID:** [RF-13]

**TÍTULO:** Acesso ao Calendário de Descanso.

**DESC:** Cada usuário deverá ser capaz de acessar as informações geradas pelo sistema nos [RF-07], [RF-08], [RF-09].

Um usuário do tipo “funcionário” só será capaz de acessar informações referentes a ele mesmo: datas nas quais receberá folga, e histórico de folgas.

Usuários do tipo “gerente” poderão acessar todas as informações referentes à folga de todos os funcionários.

**RAT:** Útil para facilitar a gestão dos funcionários.

**DEP:** [RF-04], [RF-07], [RF-08], [RF-09].

**PRI:** Altíssima.

# Apêndice 5 - Requisitos Não Funcionais

As abreviações utilizadas são: ID (Identificação), RNF (Requisito Não-Funcional), DESC (Descrição), RAT (Racionalidade, motivo), DEP (Dependências), e PRI (Prioridade).

**ID:** [RNF-01]

**TÍTULO:** Desempenho.

**DESC:** As diferentes funções do sistema devem ser processadas o mais rápido possível pelo servidor, com o fim de garantir um menor tempo de espera do usuário. Para garantir isso, o sistema deverá ter interfaces simples e responsivas.

**RAT:** Necessário para garantir uma boa experiência de usuário, e um gasto menor de recursos computacionais.

**DEP:** Nenhuma.

**PRI:** Alta.

**ID:** [RNF-02]

**TÍTULO:** Manutenabilidade.

**DESC:** O sistema deve ser projetado e implementado de forma a permitir fácil manutenção e escalabilidade. Deve ser possível adicionar novas funcionalidades ou ajustar as existentes sem grandes dificuldades.

**RAT:** Garantir que o sistema possa crescer e se adaptar às necessidades futuras da empresa.

**DEP:** Nenhuma.

**PRI:** Alta.

**ID:** [RNF-03]

**TÍTULO:** Disponibilidade.

**DESC:** O sistema deve estar disponível para uso dos usuários durante a maior parte do tempo, com o mínimo possível de períodos de inatividade não planejados.

**RAT:** Essencial para garantir que os usuários possam acessar o sistema sempre que necessário.

**DEP:** Nenhuma.

**PRI:** Alta.

**ID:** [RNF-04]

**TÍTULO:** Integridade.

**DESC:** O banco de dados deve ser projetado para garantir a integridade dos dados, evitando perdas ou corrupção de informações.

**RAT:** Fundamental para assegurar a confiabilidade e precisão das informações armazenadas.

**DEP:** Nenhuma.

**PRI:** Alta.

**ID:** [RNF-05]

**TÍTULO:** Segurança.

**DESC:** O sistema deve implementar medidas robustas de segurança para proteger os dados dos usuários, incluindo criptografia de dados, controle de acesso e prevenção de ataques cibernéticos.

**RAT:** Essencial para proteger informações sensíveis e garantir a confiança dos usuários no sistema.

**DEP:** [RNF-03], [RNF-04], [RNF-08].

**PRI:** Altíssima.

**ID:** [RNF-06]

**TÍTULO:** Usabilidade.

**DESC:** O sistema deve ter uma interface intuitiva e fácil de usar, com uma experiência de usuário agradável e eficiente.

**RAT:** Importante para garantir a satisfação dos usuários com o sistema.

**DEP:** [RNF-09].

**PRI:** Alta.

**ID:** [RNF-07]

**TÍTULO:** Baixo Custo.

**DESC:** O desenvolvimento e a manutenção do sistema devem ser feitos de forma econômica, utilizando ferramentas e tecnologias acessíveis e minimizando os custos operacionais.

**RAT:** Importante para garantir a viabilidade financeira do projeto.

**DEP:** Nenhuma.

**PRI:** Média.

**ID:** [RNF-08]

**TÍTULO:** Confidencialidade.

**DESC:** As informações dos usuários devem ser tratadas com confidencialidade e não devem ser acessíveis a pessoas não autorizadas.

**RAT:** Fundamental para proteger a privacidade e os dados sensíveis dos usuários.

**DEP:** Nenhuma.

**PRI:** Altíssima.

**ID:** [RNF-09]

**TÍTULO:** Interface Simples e Intuitiva.

**DESC:** A interface do sistema deve ser projetada de forma simples e intuitiva, facilitando a interação dos usuários com as funcionalidades do sistema.

**RAT:** Importante para garantir uma curva de aprendizado reduzida e uma experiência de usuário positiva.

**DEP:** Nenhuma.

**PRI:** Alta.

**ID:** [RNF-10]

**TÍTULO:** Tempo Mínimo de Produção do Produto.

**DESC:** O tempo necessário para desenvolver e implementar o sistema deve ser minimizado, sem comprometer a qualidade e a eficiência do produto final.

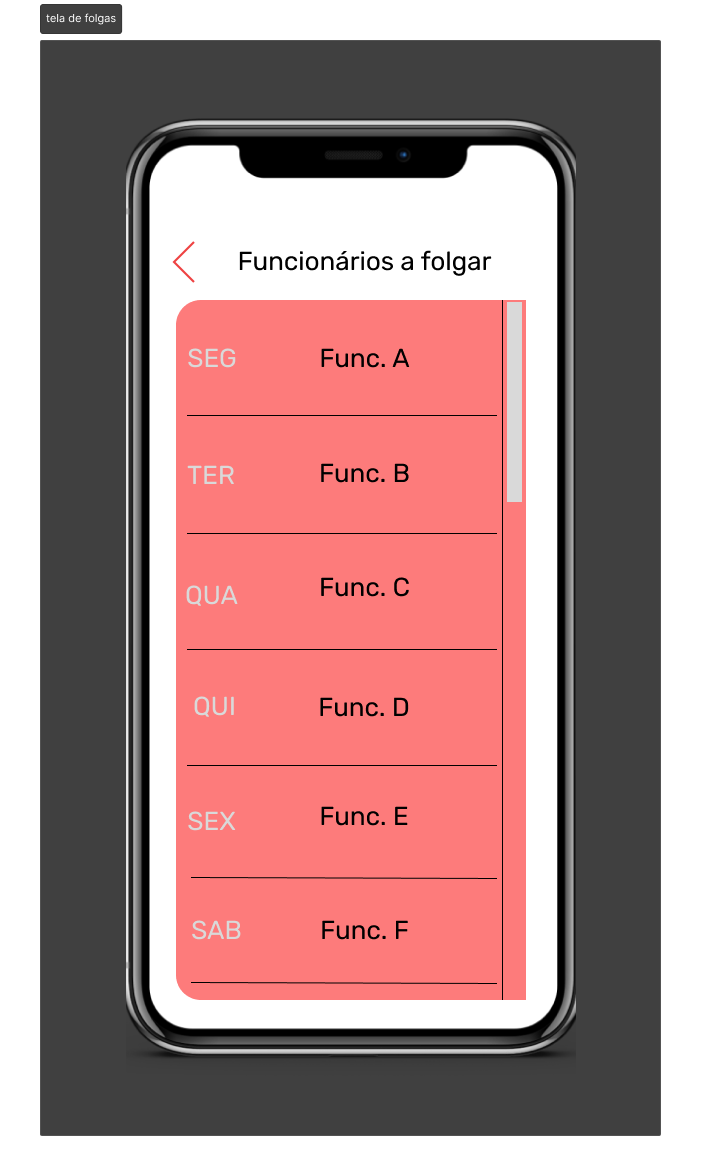
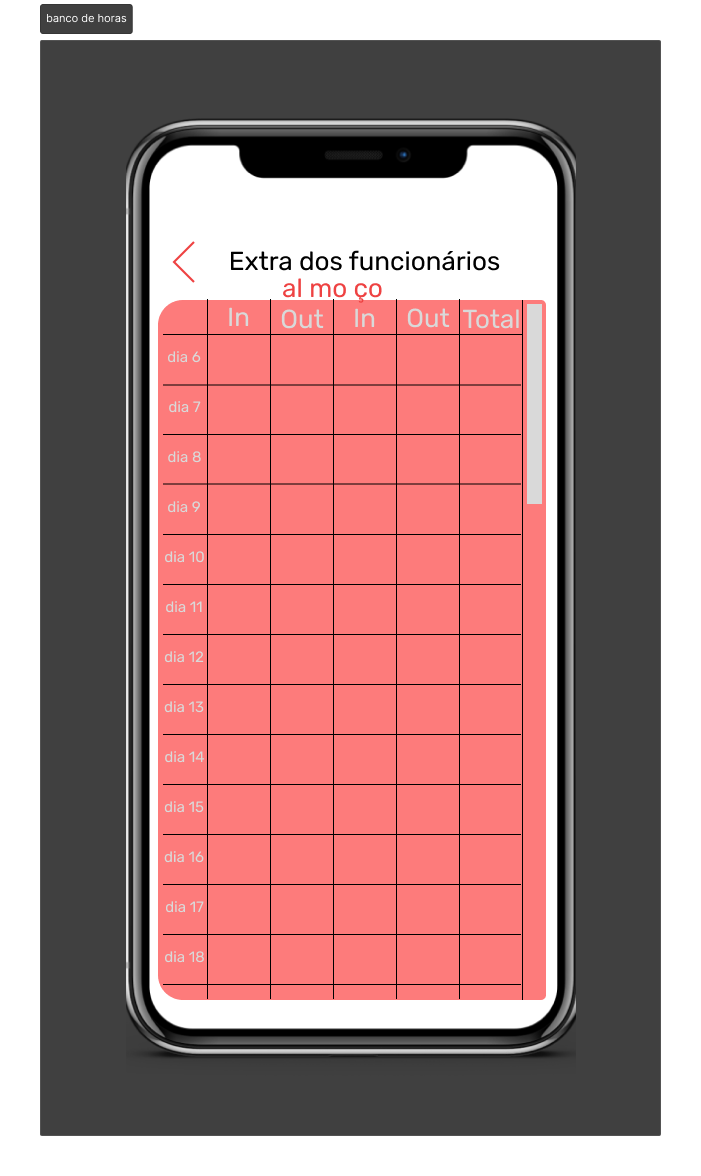
**RAT:** Importante para garantir que o sistema seja lançado no mercado dentro do prazo estipulado.

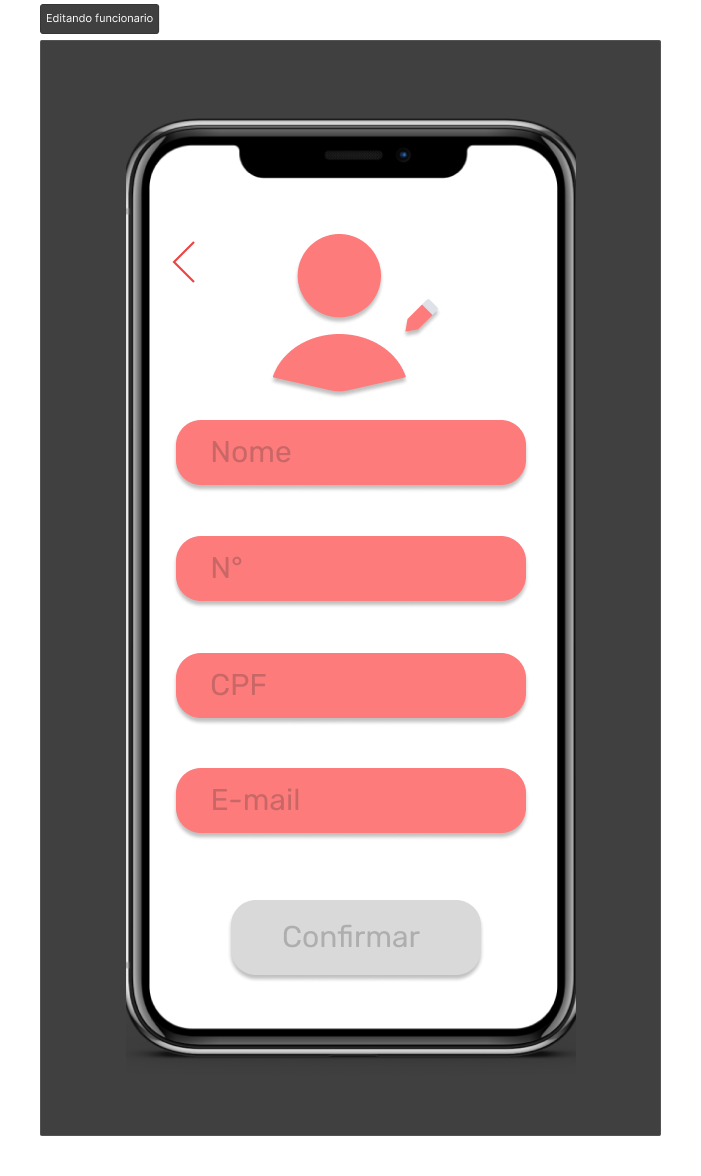
**DEP:** Nenhuma.

**PRI:** Média.

# Apêndice 6 - PROTÓTIPOS PRELIMINARES DA INTERFACE

# 

**  **

**  **