

UNIVERSIDADE ESTADUAL DE PONTA GROSSA  
SETOR DE ENGENHARIAS, CIÊNCIAS AGRÁRIAS E DE TECNOLOGIA  
DEPARTAMENTO DE INFORMÁTICA  
ENGENHARIA DE COMPUTAÇÃO

MARCOS RENAN KRUL  
RENATO CRISTIANO RUPPEL

PROJETO DE SOFTWARE: PSIS.

PONTA GROSSA

2022

MARCOS RENAN KRUL  
RENATO CRISTIANO RUPPEL

PROJETO DE SOFTWARE: PSIS.

Desenvolvimento do projeto para a disciplina de Projeto de Software do curso de Engenharia de Computação da Universidade Estadual de Ponta Grossa.

Docentes: Diolete Marcante Lati Cerutti,  
Ezequiel Gueiber, Idomar Augusto Cerutti.

PONTA GROSSA

2022

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>9</b>
<b>2</b>	<b>DIAGNÓSTICO ATUAL .....</b>	<b>10</b>
<b>3</b>	<b>OBJETIVO DO PROJETO .....</b>	<b>11</b>
<b>4</b>	<b>ESCOPO .....</b>	<b>12</b>
<b>5</b>	<b>PROPOSTA DO PROJETO .....</b>	<b>13</b>
5.1	BENEFÍCIOS .....	13
5.2	CUSTOS .....	14
5.3	RISCOS .....	17
5.4	DESCRIÇÃO DA MODELAGEM .....	19
5.4.1	Modelagem orientada a objetos .....	19
5.4.1.1	Diagrama de casos de uso .....	19
5.4.1.2	Diagrama de classes .....	20
5.4.2	Modelagem do banco de dados .....	22
5.4.2.1	DER .....	22
5.4.2.2	IDEF1X .....	23
5.4.2.4	Dicionário de Dados .....	24
<b>6</b>	<b>INTERFACE DO USUÁRIO .....</b>	<b>32</b>
<b>7</b>	<b>PLANO DE TESTES .....</b>	<b>42</b>
<b>8</b>	<b>SCRIPT PARA CRIAÇÃO DO BANCO DE DADOS .....</b>	<b>49</b>
<b>9</b>	<b>CRONOGRAMA .....</b>	<b>57</b>
<b>10</b>	<b>CONCLUSÃO .....</b>	<b>58</b>
<b>11</b>	<b>REFERÊNCIAS .....</b>	<b>59</b>
<b>12</b>	<b>RESPONSABILIDADES .....</b>	<b>60</b>

## LISTA DE FIGURAS

FIGURA 1	- Diagrama de casos de uso .....	20
FIGURA 2	- Diagrama de classes do estereótipo Entidade .....	21
FIGURA 3	- Diagrama de classes do estereótipo Controlador .....	21
FIGURA 4	- Diagrama de classes do estereótipo Fronteira .....	22
FIGURA 5	- Modelagem conceitual entidade relacionamento.....	23
FIGURA 6	- Modelagem lógica IDEF1X .....	24
FIGURA 7	- Tela de Login .....	32
FIGURA 8	- Tela da agenda .....	32
FIGURA 9	- Tela da agenda em modo semanal .....	33
FIGURA 10	- Tela da agenda em modo mensal .....	33
FIGURA 11	- Tela de visualização de uma consulta com <i>status</i> agendado ..	34
FIGURA 12	- Tela de visualização de uma consulta com <i>status</i> confirmado na visão do funcionário .....	34
FIGURA 13	- Tela de visualização de uma consulta com <i>status</i> confirmado na visão do profissional .....	35
FIGURA 14	- Tela de visualização de uma consulta com <i>status</i> concluído ...	35
FIGURA 15	- Tela de escolha de um espaço vazio para agendamento .....	36
FIGURA 16	- Tela de agendamento com dados do paciente preenchidos ...	36
FIGURA 17	- Tela de criação de restrição de horário na agenda .....	37
FIGURA 18	- Tela de criação de anotação após conclusão da consulta .....	37
FIGURA 19	- Tela de listagem de pacientes .....	38
FIGURA 20	- Tela de cadastro de paciente .....	38
FIGURA 21	- Tela de listagem de consultas concluídas de um paciente .....	39
FIGURA 22	- Tela de configurações de agenda do profissional .....	39
FIGURA 23	- Tela da agenda com a ajuda referente às cores dos eventos ..	40
FIGURA 24	- Tela da agenda com a ajuda referente ao uso para colaboradores .....	41

FIGURA 25	- Tela de configurações de agenda do profissional com a ajuda aberta .....	41
-----------	--	----

## LISTA DE QUADROS

QUADRO 1	- Descrição do risco relacionado à desistência do cliente .....	18
QUADRO 2	- Descrição do risco relacionado ao atraso .....	18
QUADRO 3	- Descrição do risco relacionado às tecnologias .....	19
QUADRO 4	- Descrição dos atributos da tabela responsável pela entidade <i>Clinica</i> .....	24
QUADRO 5	- Descrição das restrições da tabela responsável pela entidade <i>Clinica</i> .....	25
QUADRO 6	- Descrição dos atributos da tabela responsável pela entidade <i>Pessoa</i> .....	25
QUADRO 7	- Descrição das restrições da tabela responsável pela entidade <i>Pessoa</i> .....	25
QUADRO 8	- Descrição dos atributos da tabela responsável pela entidade <i>Usuario</i> .....	26
QUADRO 9	- Descrição dos atributos da tabela responsável pela entidade <i>Profissional</i> .....	26
QUADRO 10	- Descrição das restrições da tabela responsável pela entidade <i>Profissional</i> .....	26
QUADRO 11	- Descrição dos atributos da tabela responsável pela entidade <i>Paciente</i> .....	27
QUADRO 12	- Descrição das restrições da tabela responsável pela entidade <i>Paciente</i> .....	27
QUADRO 13	- Descrição dos atributos da tabela responsável pela entidade <i>Endereco</i> .....	27
QUADRO 14	- Descrição dos atributos da tabela responsável pela entidade <i>Responsavel</i> .....	28
QUADRO 15	- Descrição dos atributos da tabela responsável pela entidade <i>AgendaSemanal</i> .....	28
QUADRO 16	- Descrição das restrições da tabela responsável pela entidade <i>AgendaSemanal</i> .....	28
QUADRO 17	- Descrição dos atributos da tabela responsável pela entidade <i>RestricoesHorario</i> .....	29
QUADRO 18	- Descrição das restrições da tabela responsável pela entidade <i>RestricoesHorario</i> .....	29

QUADRO 19	- Descrição dos atributos da tabela responsável pela entidade <i>RestricoesAgendaSemanal</i> .....	29
QUADRO 20	- Descrição das restrições da tabela responsável pela entidade <i>RestricoesAgendaSemanal</i> .....	29
QUADRO 21	- Descrição dos atributos da tabela responsável pela entidade <i>Consulta</i> .....	30
QUADRO 22	- Descrição das restrições da tabela responsável pela entidade <i>Consulta</i> .....	30
QUADRO 23	- Descrição dos atributos da tabela responsável pela entidade <i>PapelUsuario</i> .....	31
QUADRO 24	- Descrição dos atributos da tabela responsável pela entidade <i>Permissoes</i> .....	31
QUADRO 25	- Descrição do plano de testes para o módulo referentes às consultas .....	42
QUADRO 26	- Descrição do plano de testes para o módulo referentes às operações básicas sobre a entidade <i>Paciente</i> .....	43
QUADRO 27	- Descrição do plano de testes para o módulo referentes às operações sobre a agenda dos profissionais .....	45

## LISTA DE TABELAS

TABELA 1	- Cálculo do peso não ajustado por atores .....	14
TABELA 2	- Cálculo do peso não ajustado por casos de uso .....	15
TABELA 3	- Cálculo fator de ajuste para complexidades técnicas .....	15
TABELA 4	- Cálculo fator de ajuste para complexidades do ambiente .....	16
TABELA 5	- Cálculo das variáveis para análise por pontos de caso de uso.	17



## 1 INTRODUÇÃO

O trabalho aqui desenvolvido, denominado projeto de *software*, tem o intuito de descrever todos os pontos necessários para a criação de um produto de sistema computacional direcionado à psicóloga Renata Adriele Ruppel, a qual foi consultada sobre as características que este deve possuir, a partir de obstáculos existentes no processo atual. Diante disso, serão modelados todas as características necessárias para a compreensão do projeto, facilitando o início do desenvolvimento do mesmo.

## 2 DIAGNÓSTICO ATUAL

Este *software* se destina à profissional liberal psicóloga Renata Adriele Ruppel, atualmente servidora pública em São José dos Pinhais, Paraná, que irá utilizar o mesmo no retorno de seus atendimentos em uma clínica de atendimentos psicológicos com vários profissionais, localizada em Curitiba, Paraná, dando continuidade às ações realizadas anteriormente em uma clínica localizada em Irati, Paraná.

O processo de agendamento para os profissionais da clínica é realizado por meio de contato direto dos pacientes com os atendentes ou com os clínicos, utilizando meios de comunicação tradicionais (telefone) ou pessoais (*WhatsApp*). Após o contato, as informações pertinentes são armazenadas em uma agenda física de papel. Esta ferramenta não apresenta um bom desempenho, visto que não há um bom gerenciamento e organização em seu uso, gerando confusões durante o fluxo de trabalho. O processo foi definido pelo cliente como custoso, que indo além, relatou o fluxo: O paciente entra em contato para marcar uma consulta, o atendente precisa checar os dias e horários disponíveis do profissional (que são disponibilizados a ele em uma agenda física), tenta encaixar o horário compatível do paciente com o profissional para a consulta, e, ao fim do dia, o clínico revisa seus horários marcados naquele dia para garantir a integridade.

Assim como os horários, as anotações sobre cada consulta e paciente também eram mantidas em uma agenda de papel, gerando os mesmos problemas inerentes ao uso dessa ferramenta, como a perda de informações e dificuldade na recuperação e visualização das mesmas.

A partir do contexto e do fluxo apresentado, levantou-se os problemas que fazem com que o uso da agenda de papel seja um empecilho, como a falta de confirmação das consultas, ausência ou perda de dados dos pacientes e falta de especificidade, como, por exemplo, consultas para crianças que não eram identificadas como tal.

### **3 OBJETIVO DO PROJETO**

#### **a) Objetivo geral**

Desenvolver um sistema computacional para facilitar e otimizar o processo de agendamento de consultas para cada profissional em uma clínica médica psicológica.

#### **b) Objetivos específicos**

- Garantir consistência nos agendamentos, como cancelamentos e a confirmação das consultas;
- Criar mecanismos para informar disponibilidade de horários na agenda de cada profissional;
- Facilitar a recuperação e visualização de informações a respeito dos agendamentos e dos pacientes;
- Garantir a obrigatoriedade das informações necessárias; e
- Criar mecanismos para manter anotações dos profissionais para os pacientes a cada consulta.

## **4 ESCOPO**

O sistema irá proporcionar um controle e gerenciamento do agendamento de consultas para cada profissional da clínica. Para isso, cada clínico, em seu acesso particular, poderá definir seus horários disponíveis na agenda semanal, propagando-os para as semanas subsequentes, e horários que não poderá atender, sendo estes datas ou horários específicos. Além disso, o profissional poderá ter um controle particular sobre as suas anotações para cada consulta de todos os seus pacientes.

O sistema desenvolvido terá que cobrir o armazenamento e gerenciamento de profissionais, funcionários, pacientes e administradores. Será possível, ainda, realizar as outras operações básicas, como atualização, deleção e recuperação, controladas a partir do acesso baseado em permissões, ou seja, cada grupo de usuários terá seu papel no sistema.

Não entra no escopo do sistema computacional o gerenciamento de valores e recebimentos por profissionais ou pela clínica. Também não será contemplada a coleta de informações de pacientes por meio de anamnese.

## 5 PROPOSTA DO PROJETO

Concepção e desenvolvimento de um sistema computacional *web* para uma clínica, que contemplará o gerenciamento de funcionários, profissionais e pacientes, bem como os horários destes profissionais e o agendamento de consultas. Para as aplicações, serão utilizados:

- A linguagem *Javascript: ECMAScript 2018 (superset Typescript: 4.6.3)*;
- O *framework Node JS: 16.15.1* para *back-end*;
- O *framework React JS: 18.0.0* para o *front-end web*;
- O SGBD *PostgreSQL: 14.1* para banco de dados.

Para a utilização de recursos em nuvem nas fases de homologação e produção, como o banco de dados e serviços de *back-end* e *front-end web*, a plataforma *Amazon Web Services (AWS)* será utilizada em seu regime gratuito.

### 5.1 BENEFÍCIOS

Espera-se, ao desenvolver e implantar o sistema, prover ao cliente um ambiente que facilite e otimize o processo de agendamento de consultas e o gerenciamento de informações importantes de profissionais, funcionários e pacientes. O sistema, portanto, permitirá uma economia de tempo, pela facilidade no armazenamento e na recuperação dos dados, redução de erros relacionados às falhas do uso de uma agenda física, e uma maior segurança das informações, tanto pela redução de perdas quanto pelo sigilo imposto através do acesso baseado em permissões.

Por parte dos clínicos, espera-se que haja um ganho nos aspectos de planejamento e controle da sua agenda semanal, a partir do acesso à sua agenda particular. Ainda, os mesmos poderão utilizar os mecanismos referentes ao controle das anotações de cada consulta para seus pacientes, facilitando a recuperação e visualização das mesmas, com o benefício, ainda, relacionado à segurança, tanto pela redução de perdas quanto pelo sigilo necessário.

## 5.2 CUSTOS

Dado o problema de precificação de um *software*, pode-se utilizar métricas orientadas a casos de uso, os quais são usados para descrever os requisitos que propõe as funções do software. Em comparação a outros métodos, possui benefícios como a abstração das tecnologias utilizadas, visto que o diagrama de casos de uso é independente de linguagens de programação, ou seja, pode-se estimar o custo de implementação na fase inicial do projeto, sem que haja a necessidade de iniciar a codificação. Existe, contudo, uma incerteza inerente ao resultado obtido através desta técnica, associada à falta de padronização do tamanho de cada caso de uso, relacionado, também, ao nível que o diagrama foi desenvolvido. (1), (3), (4)

Para o levantamento do custo, por fim, foi realizada uma pesquisa para entender e estipular uma média salarial de um programador, levando em consideração parâmetros como região, nível de conhecimento, experiência de mercado, carga horária e porte de empresas do ramo. Com isso, foi definido o valor de R\$ 12,85 por horas trabalhadas para um programador Júnior, com perfil para um programador de 22 anos, residente na cidade de Ponta Grossa, Paraná, em empresas de pequeno ou médio porte. (2)

TABELA 1 - Cálculo do peso não ajustado por atores.

<b><i>Unadjusted Actor Weight</i></b>				
Tipo de ator	Peso	Descrição	Quantidade	Resultado
Simple	1	Outro sistema acessado através de uma API de programação	1	1
Médio	2	Outro sistema acessado interagindo através da rede	0	0
Complexo	3	Um usuário interagindo através de uma interface gráfica	3	9
<b>TOTAL UAW</b>			4	10

Fonte: Os autores.

TABELA 2 - Cálculo do peso não ajustado por casos de uso.

<b><i>Unadjusted Use Case Weight</i></b>				
Tipo Use Case	Peso	Descrição	Quantidade	Resultado
Simple	5	Tem até 3 transações, incluindo os passos alternativos, e envolve menos de 5 entidades	7	35
Médio	10	Tem de 4 a 7 transações, incluindo os passos alternativos, e envolve de 5 a 10 entidades	3	30
Complexo	15	Tem acima de 7 transações, incluindo os passos alternativos, e envolve pelo menos de 10 entidades	0	0
<b>TOTAL UUCW</b>			10	65

Fonte: Os autores.

TABELA 3 - Cálculo fator de ajuste para complexidades técnicas.

<b><i>Technical Complexity Factor</i></b>				
Fator	Peso	Descrição	Influência	Resultado
T1	2	Sistema distribuído	1	2
T2	2	Tempo de resposta	2	4
T3	1	Eficiência	3	3
T4	1	Processamento complexo	2	2
T5	1	Código reusável	3	3
T6	0,5	Facilidade de instalação	2	1
T7	0,5	Facilidade de uso	2	1
T8	2	Portabilidade	1	2
T9	1	Facilidade de mudança	2	2
T10	1	Concorrência	2	2
T11	1	Recursos de segurança	2	2
T12	1	Acessível por terceiros	0	0
T13	1	Requer treinamento especial	1	1
<b>TOTAL Tfactor</b>			23	25

Fonte: Os autores.

TABELA 4 - Cálculo fator de ajuste para complexidades do ambiente.

<b><i>Environmental Complexity Factor</i></b>				
Fator	Peso	Descrição	Influência	Resultado
E1	1,5	Familiaridade com RUP ou outro processo formal	3	4,5
E2	0,5	Experiência com a aplicação em desenvolvimento	5	2,5
E3	1	Experiência em orientação a objetos	4	4
E4	0,5	Capacidade dos analistas	2	1
E5	1	Motivação	5	5
E6	2	Requisitos estáveis	5	10
E7	-1	Desenvolvedores em meio-expediente	3	-3
E8	-1	Linguagem de programação difícil	0	0
<b>TOTAL Efactor</b>			27	24

Fonte: Os autores.

A Tabela 5 apresenta os cálculos finais para o método dos pontos por casos de uso, para isso, utiliza os resultados das Tabelas 1, 2, 3 e 4, e também variáveis externas, como a carga horária semanal de trabalho e o custo por hora. A partir do nível de conhecimento dos membros e da análise geral do problema, a maior parte dos custos do projeto será no âmbito de mão de obra interna, relacionada à jornada de trabalho de cada membro da equipe para realizar a codificação do sistema. Assim, de acordo com a técnica explicada anteriormente, com a estimativa salarial, com o diagrama de casos de uso desenvolvido e considerando uma carga semanal de 10 horas, é possível realizar os cálculos para definir o tempo de trabalho e o valor final, como mostrado na Tabela 5. (1), (3), (4)

Posterior à fase de desenvolvimento, o sistema deve entrar no estado público para que possa ser utilizado pelos usuários. Para isso, além de outros, torna-se necessária a aquisição de um domínio, o qual é um nome utilizado para identificar os sistemas na *internet* global. Dessa forma, obteve-se o domínio *psis.net.br* com o custo de R\$ 40,00 (quarenta reais) na associação responsável pelas emissões de registros brasileiros - Núcleo de Informação e Coordenação do Ponto BR (nic.br) - pelo período de um ano.



TABELA 5 - Cálculo das variáveis para análise por pontos de casos de uso.

Variável	Equação	Resultado Obtido
UUCP	$UUCP = UAW + UUCW$	75
TCF	$TCF = 0.6 + (0.01 \times Tfactor)$	0,85
ECF	$ECF = 1.4 + (-0.03 \times Efactor)$	0,68
UCP	$UCP = UUCP \times TCF \times ECF$	43,35
Tempo estimado	Tempo estimado = UCP x carga horária semanal	433,5 horas
Tempo individual	Tempo individual = Tempo estimado / 2	216,75 horas
Valor total	Valor total = tempo estimado x custo por hora	R\$ 5570,48

Fonte: Os autores.

### 5.3 RISCOS

Após uma avaliação do plano do projeto, foram levantados e estudados os riscos a que este está submetido. Vale ressaltar que há uma grande possibilidade dessa classificação não contemplar todos os riscos, visto que existem riscos imprevisíveis que podem ou não ocorrer e é extremamente difícil identificá-los com antecedência, tornando este um processo impossível. (1)

É interessante, na análise dos riscos, quantificar o nível de incerteza e o grau de perdas associadas a cada risco. Para isso, em um primeiro momento, é importante categorizá-los de acordo com a sua natureza. Os riscos de projeto colocam em ameaça o plano do projeto, podendo gerar atrasos no cronograma previamente estipulado. Os riscos técnicos, por sua vez, influenciam na qualidade e na data de entrega do *software*, dificultando ou impossibilitando a implementação do mesmo. Por fim, os riscos de negócio colocam em risco a viabilidade e, até mesmo, o projeto como um todo. (1)

Os Quadros 1, 2 e 3 apresentam os riscos levantados para o projeto, de acordo com os métodos de análise supracitados.

QUADRO 1 - Descrição do risco relacionado à desistência do cliente.

<b>Risco</b>	Desistência do cliente
<b>Categoria</b>	Risco de negócio
<b>Probabilidade</b>	5%
<b>Descrição</b>	Em qualquer etapa do projeto o cliente pode vir a desistir, anulando todo o planejamento e trabalho até aquele momento.
<b>Prevenção</b>	Bom planejamento e uma definição correta do escopo do projeto para que as expectativas sejam atendidas e, ao decorrer do prazo, serão feitos repasses em intervalos curtos de tempo, tanto para validação quanto para a tentativa de aproximar mais o cliente, evitando que o mesmo fique muito tempo sem <i>feedbacks</i> e acabe perdendo o interesse.
<b>Tratativa</b>	Tentativa de renegociação de alguns aspectos do projeto, tentando entender os motivos pela tomada desta decisão e levando em consideração se esta ocorreu devido a algum problema relacionado ao projeto, tornando possível a renegociação.

Fonte: Os autores.

QUADRO 2 - Descrição do risco relacionado ao atraso.

<b>Risco</b>	Atrasos na implementação
<b>Categoria</b>	Risco de projeto
<b>Probabilidade</b>	30%
<b>Descrição</b>	Diante de barreiras como a grande carga de trabalho, pouca experiência dos membros, dificuldades técnicas e outros problemas, há a possibilidade de que ocorram alterações no cronograma previamente estabelecido.
<b>Prevenção</b>	Definir um cronograma que seja possível de ser realizado, levando em consideração parâmetros como nível de conhecimento dos membros, nível de complexidade do projeto, carga horária de trabalho semanal, entre outros.
<b>Tratativa</b>	Em casos de atrasos, o cronograma será adaptado a fim de corrigir as falhas, executando tarefas em paralelo ou aumentando a carga horária de trabalho.

Fonte: Os autores.

QUADRO 3 - Descrição do risco relacionado às tecnologias.

<b>Risco</b>	As tecnologias escolhidas não atingirão as expectativas
<b>Categoria</b>	Risco técnico
<b>Probabilidade</b>	35%
<b>Descrição</b>	O sistema deve responder aos usuários com tempo de espera mínimo, sem que haja um congestionamento de tarefas a serem realizadas, melhorando a experiência de uso do mesmo.
<b>Prevenção</b>	Realizar um estudo das possíveis tecnologias de acordo com os requisitos previamente levantados e modelar o sistema visando a facilidade na persistência e no acesso dos dados.
<b>Tratativa</b>	Usar técnicas de análise de algoritmos para encontrar pontos de lentidão e corrigi-los, além de rever a modelagem do sistema para tentar facilitar as manipulações dos dados.

Fonte: Os autores.

## 5.4 DESCRIÇÃO DA MODELAGEM

### 5.4.1 Modelagem orientada a objetos

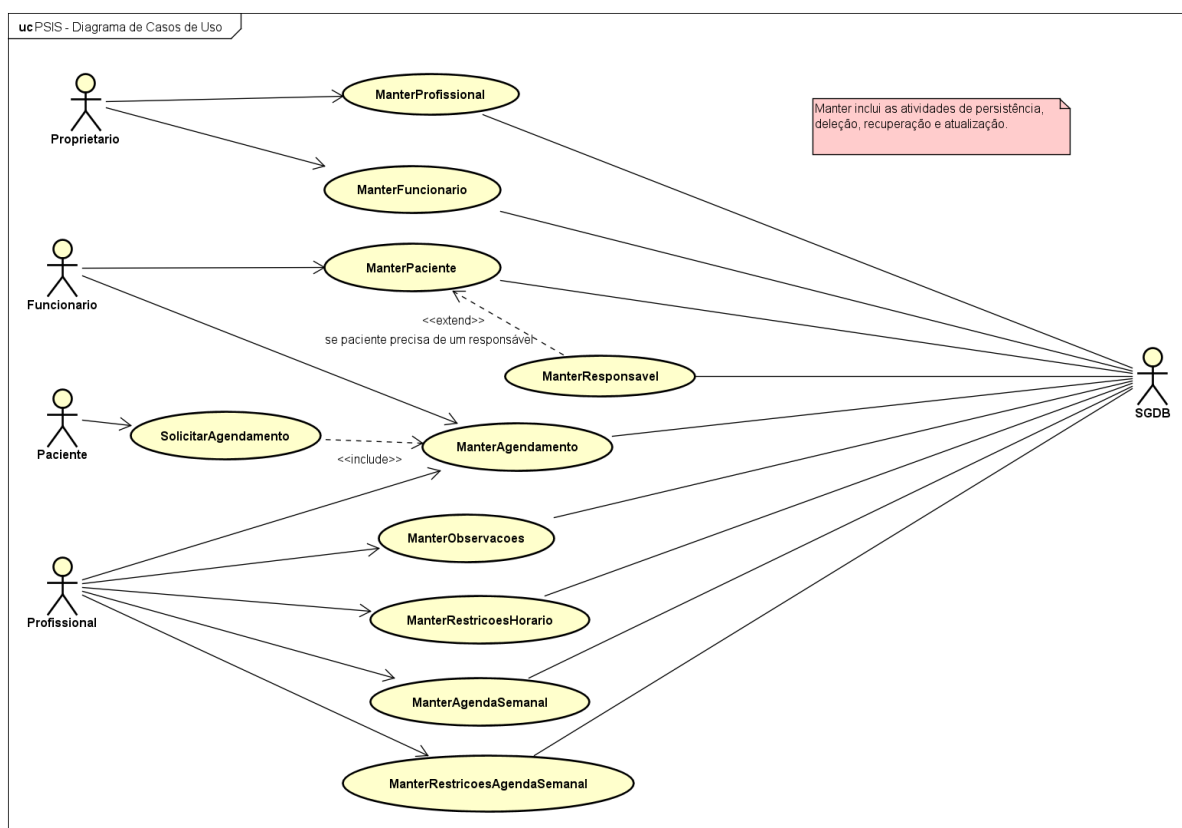
#### 5.4.1.1 Diagrama de casos de uso

Considerando os atores que irão interagir com o sistema computacional, os casos de uso, demonstrados na Figura 1, foram levantados para extração e levantamento de requisitos, demonstrando as aplicações ou usos completos do sistema, ou seja, funcionalidade do mesmo. Para isso, são considerados os quatro principais autores, cada um com seus respectivos papéis dentro do contexto do sistema a ser desenvolvido.

O primeiro ator, denominado como *Proprietario*, representa o responsável do estabelecimento, e a este são atribuídas as tarefas de gestão dos colaboradores e profissionais - casos de uso denominados como, respectivamente, *ManterFuncionario* e *ManterProfissional*. O segundo ator, denominado *Funcionario*, é responsável por iniciar os casos de uso para gerenciamento de pacientes (*ManterPaciente*), incluindo o controle de responsáveis, se necessário, além de realizar os agendamentos das consultas - caso de uso denominado *ManterAgendamento*. O último ator que representa um usuário com acesso ao sistema, por fim, denominado como *Profissional*, possui as responsabilidades de

manipulação das observações de cada consulta e gerenciamento dos horários da agenda, representado pelo casos de uso *ManterAgendaSemanal* e *ManterRestricoesAgendaSemanal* (horários livres e bloqueios que serão propagados para todas as semanas, até que hajam alterações), além do controle de restrições específicas em qualquer data futura - caso de uso denominado *ManterRestricoesHorario*.

FIGURA 1 - Diagrama de casos de uso.



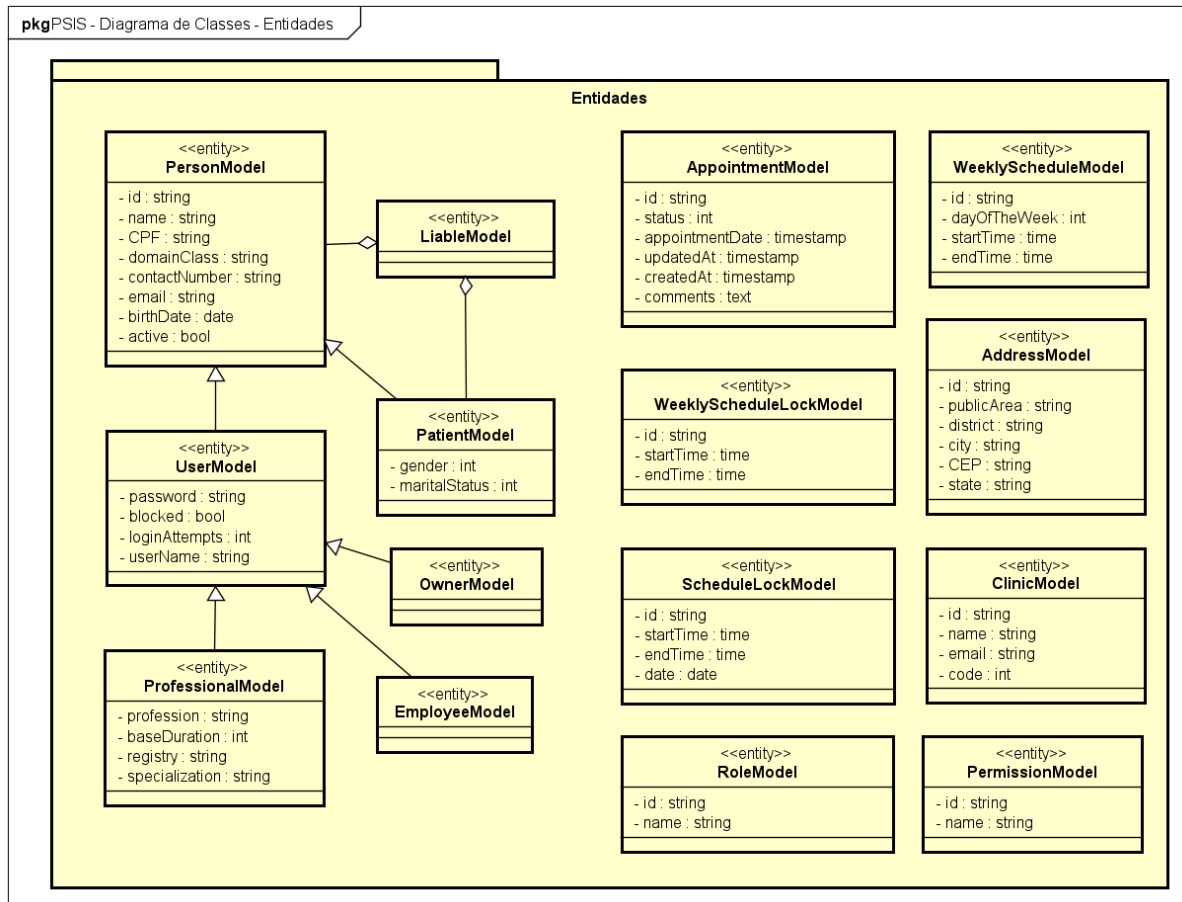
Fonte: Os autores.

#### 5.4.1.2 Diagrama de classes

O diagrama de classes, nesta etapa, foi construído levando em consideração tanto classes principais quanto auxiliares, relacionamentos entre as mesmas e atributos das entidades. Para melhor visualização, os três principais estereótipos - entidade, controlador e fronteira - foram separados em módulos

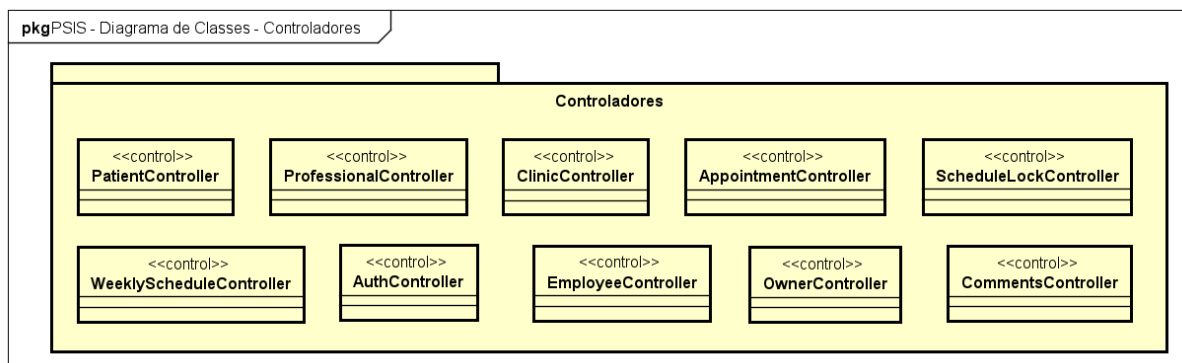
diferentes. Assim, as classes referentes às entidades são apresentadas na Figura 2, referentes aos controladores na Figura 3 e referente às fronteiras na Figura 4.

FIGURA 2 - Diagrama de classes do estereótipo Entidade.



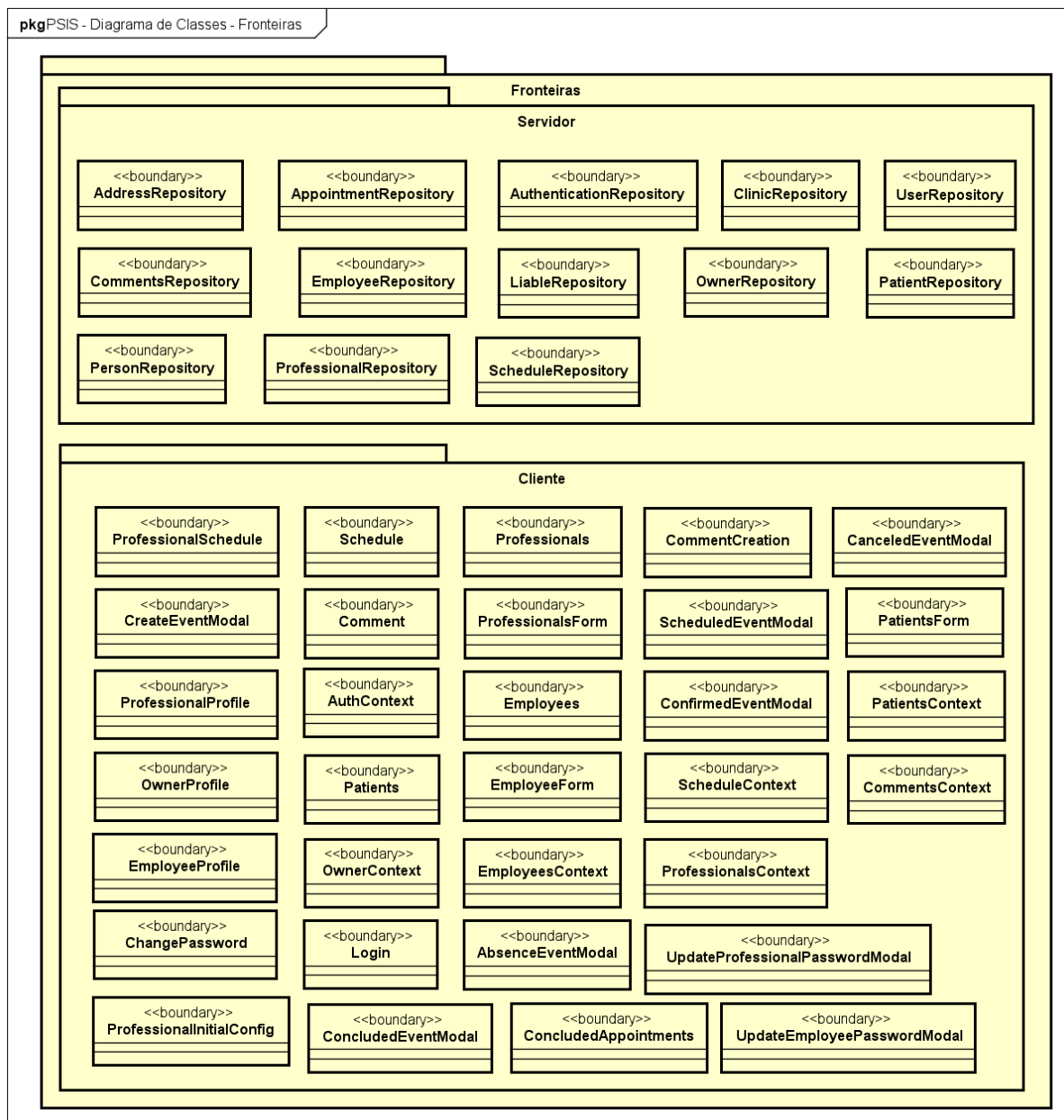
Fonte: Os autores.

FIGURA 3 - Diagrama de classes do estereótipo Controlador.



Fonte: Os autores.

FIGURA 4 - Diagrama de classes do estereótipo Fronteira.



Fonte: Os autores.

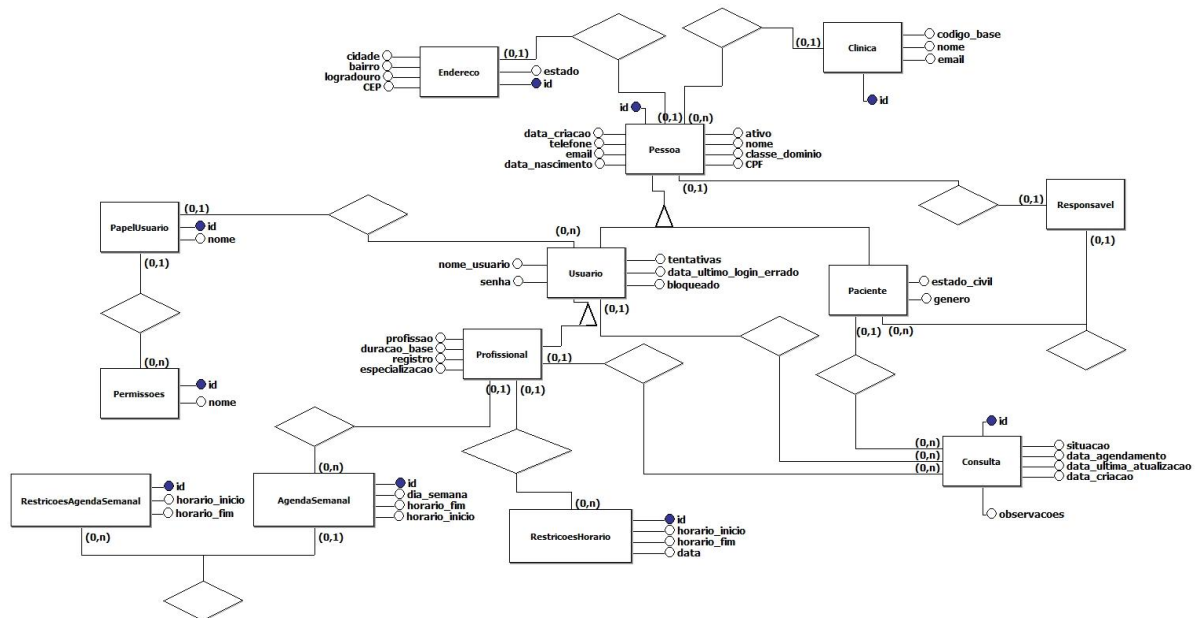
## 5.4.2 Modelagem do banco de dados

### 5.4.2.1 DER

A Figura 5 demonstra a modelagem conceitual com entidades, relações e atributos que constarão na base de dados da aplicação. Modelado a partir da análise dos requisitos do *software*, com o auxílio da validação do cliente,

principalmente para os atributos necessários de cada entidade, além do refinamento e validação para que este sirva o fluxo imaginado pela aplicação.

FIGURA 5 - Modelagem conceitual entidade relacionamento.



Fonte: Os autores.

#### 5.4.2.2 IDEF1X

A Figura 6 representa o modelo lógico, fiel à modelagem conceitual, também mostrando entidades, relações e respectivos atributos. Criado a partir da modelagem DER (Figura 5), apresentando uma descrição mais precisa dos atributos, e também todas as relações do DER que se tornarão entidades na implementação do banco de dados.





QUADRO 5 - Descrição das restrições da tabela responsável pela entidade *Clinica*.

Nome	Tipo	Definição	Descrição
clinica_email_key	Unique	(email)	E-mail deve ser único

Fonte: Os autores.

QUADRO 6 - Descrição dos atributos da tabela responsável pela entidade *Pessoa*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Não		Identificador único
nome	VARCHAR(100)	Sim	Não	Não		Nome da pessoa
cpf	VARCHAR(11)	Não	Não	Não		CPF da pessoa
email	VARCHAR(100)	Não	Não	Não		E-mail para contato
telefone	VARCHAR(11)	Sim	Não	Não		Telefone para contato
data_nascimento	DATE	Sim	Não	Não		Data de nascimento
classe_dominio	VARCHAR(32)	Sim	Não	Não		Hierarquia de classes da tupla
data_criacao	TIMESTAMP	Sim	Não	Não	CURRENT_TIMESTAMP	Data de criação do registro
ativo	BOOLEAN	Sim	Não	Não	true	Se a pessoa está ativa na clínica
id_clinica	UUID	Sim	Não	Sim		Identificação da clínica

Fonte: Os autores.

QUADRO 7 - Descrição das restrições da tabela responsável pela entidade *Pessoa*.

Nome	Tipo	Definição	Descrição
peessoa_check_nascimento	Check	data_nascimento < CURRENT_TIMESTAMP	Data de nascimento deve ser menor que a data atual

Fonte: Os autores.

QUADRO 8 - Descrição dos atributos da tabela responsável pela entidade *Usuario*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Sim		Identificador único
nome_usuario	VARCHAR(100)	Sim	Não	Não		Nome de usuário
senha	VARCHAR(255)	Sim	Não	Não		Senha do usuário
tentativas	INTEGER	Sim	Não	Não	0	Número de tentativas de <i>login</i>
bloqueado	BOOLEAN	Sim	Não	Não	false	Bloqueado para fazer login
data_ultimo_login_errado	TIMESTAMP	Não	Não	Não		Data da último <i>login</i> errado
id_role	UUID	Sim	Não	Sim		Identificador do papel do usuário

Fonte: Os autores.

QUADRO 9 - Descrição dos atributos da tabela responsável pela entidade *Profissional*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Sim		Identificador único
profissao	VARCHAR(255)	Sim	Não	Não		Profissão do clínico
duracao_base	INTEGER	Sim	Não	Não	60	Duração base das consultas (minutos)
registro	VARCHAR(255)	Sim	Não	Não		Registro médico
especializacao	VARCHAR(255)	Não	Não	Não		Especializações

Fonte: Os autores.

QUADRO 10 - Descrição das restrições da tabela responsável pela entidade *Profissional*.

Nome	Tipo	Definição	Descrição
profissional_duracao_base_check	Check	duracao_base > 0	Duração base deve ser maior que 0

Fonte: Os autores.

QUADRO 11 - Descrição dos atributos da tabela responsável pela entidade *Paciente*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Sim		Identificador único
estado_civil	SMALLINT	Sim	Não	Não		Estado civil - valor numérico inteiro entre 1 e 4, onde: 1 = Casado 2 = Divorciado 3 = Solteiro 4 = Viúvo
genero	SMALLINT	Sim	Não	Não		Gênero - valor numérico inteiro entre 1 e 5, onde: 1 = Feminino 2 = Masculino 3 = Não binário 4 = Prefiro não dizer 5 = Transgênero

Fonte: Os autores.

QUADRO 12 - Descrição das restrições da tabela responsável pela entidade *Paciente*.

Nome	Tipo	Definição	Descrição
paciente_check_genero	Check	genero IN (1,2,3,4,5)	Valor do gênero deve estar entre 1 e 5
paciente_check_estado_civil	Check	estado_civil IN (1,2,3,4)	Valor do estado civil deve estar entre 1 e 4

Fonte: Os autores.

QUADRO 13 - Descrição dos atributos da tabela responsável pela entidade *Endereco*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Não		Identificador único
bairro	VARCHAR(100)	Não	Não	Não		Bairro
logradouro	VARCHAR(255)	Sim	Não	Não		Logradouro
estado	VARCHAR(100)	Sim	Não	Não		Unidade Federativa
CEP	VARCHAR(8)	Sim	Não	Não		CEP
cidade	VARCHAR(100)	Sim	Não	Não		Cidade
id_pessoa	UUID	Sim	Não	Sim		Identificador da pessoa

Fonte: Os autores.

QUADRO 14 - Descrição dos atributos da tabela responsável pela entidade *Responsavel*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id_paciente	UUID	Sim	Sim	Sim		Identificação do paciente
id_pessoa	UUID	Sim	Não	Sim		Identificação do responsável

Fonte: Os autores.

QUADRO 15 - Descrição dos atributos da tabela responsável pela entidade *AgendaSemanal*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Não		Identificação do horário da agenda semanal
dia_semana	SMALLINT	Sim	Não	Não		Identificação do dia da semana, valor numérico inteiro entre 1 e 7, onde 1 refere-se ao domingo e 7 ao sábado.
horario_inicio	TIME	Sim	Não	Não		Horário de início
horario_fim	TIME	Sim	Não	Não		Horário de término
id_profissional	UUID	Sim	Não	Sim		Identificação do profissional

Fonte: Os autores.

QUADRO 16 - Descrição das restrições da tabela responsável pela entidade *AgendaSemanal*.

Nome	Tipo	Definição
agenda_semanal_check_dia_semana	Check	dia_semana IN (1,2,3,4,5, 6, 7)
agenda_semanal_check_horario	Check	horario_inicio < horario_fim

Fonte: Os autores.

QUADRO 17 - Descrição dos atributos da tabela responsável pela entidade *RestricoesHorario*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Não		Identificação da restrição
data	DATE	Sim	Não	Não		Data da restrição
horario_inicio	TIME	Sim	Não	Não		Horário de início
horario_fim	TIME	Sim	Não	Não		Horário de término
id_profissional	UUID	Sim	Não	Sim		Identificação do profissional

Fonte: Os autores.

QUADRO 18 - Descrição das restrições da tabela responsável pela entidade *RestricoesHorario*.

Nome	Tipo	Definição	Descrição
restricoes_horario_check_data	Check	data + horario_inicio > CURRENT_TIMESTAMP	A data da restrição deve ser futura
restricoes_horario_check_horario	Check	horario_inicio < horario_fim	Horário de término deve ser superior ao horário de início

Fonte: Os autores.

QUADRO 19 - Descrição dos atributos da tabela responsável pela entidade *RestricoesAgendaSemanal*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Não		Identificação da restrição do horário semanal para um dia da semana
horario_inicio	TIME	Sim	Não	Não		Horário de início
horario_fim	TIME	Sim	Não	Não		Horário de término
id_agenda_semanal	UUID	Sim	Não	Sim		Identificação da agenda semanal

Fonte: Os autores.

QUADRO 20 - Descrição das restrições da tabela responsável pela entidade *RestricoesAgendaSemanal*.

Nome	Tipo	Definição	Descrição
restricoes_agenda_semanal_check_horario	Check	horario_inicio < horario_fim	Horário de término deve ser superior ao horário de início

Fonte: Os autores.

QUADRO 21 - Descrição dos atributos da tabela responsável pela entidade *Consulta*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	UUID	Sim	Sim	Não		Identificador único da consulta
situacao	SMALLINT	Sim	Não	Não		Situação da consulta, valor numérico inteiro entre 1 e 5, onde: 1 = Agendado 2 = Cancelado 3 = Confirmado 4 = Cancelado 5 = Ausência
data_agendamento	TIMESTAMP	Sim	Não	Não		Data de realização da consulta (futura)
data_ultima_atualizacao	TIMESTAMP	Sim	Não	Não		Data da última atualização no registro da consulta
data_criacao	TIMESTAMP	Sim	Não	Não		Data da criação do registro da consulta
observacoes	TEXT	Não	Não	Não		Observações sobre a consulta, exclusivo para o profissional responsável
id_paciente	UUID	Sim	Não	Sim		Identificador do paciente
id_profissional	UUID	Sim	Não	Sim		Identificador do profissional/clínico
id_funcionario	UUID	Sim	Não	Sim		Identificador do funcionário responsável pela criação da consulta

Fonte: Os autores.

QUADRO 22 - Descrição das restrições da tabela responsável pela entidade *Consulta*.

Nome	Tipo	Definição	Descrição
consulta_check_situacao	Check	situacao IN (1,2,3,4,5)	Valor da situação deve estar entre 1 e 5

Fonte: Os autores.

QUADRO 23 - Descrição dos atributos da tabela responsável pela entidade *PapelUsuario*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	SERIAL	Sim	Sim	Não		Identificação do papel do usuário
nome	VARCHAR(36)	Sim	Não	Não		Nome do papel

Fonte: Os autores.

QUADRO 24 - Descrição dos atributos da tabela responsável pela entidade *Permissoes*.

Nome	Tipo	Não nulo?	PK?	FK?	Default	Descrição
id	SERIAL	Sim	Sim	Não		Identificação da permissão
nome	VARCHAR(36)	Sim	Não	Não		Nome da permissão
id_papel_usuario	INTEGER	Sim	Não	Sim		Identificador do papel de usuário

Fonte: Os autores.

## 6 INTERFACE DO USUÁRIO

As Figuras 7 à 22 apresentam os resultados das interfaces principais do sistema.

FIGURA 7 - Tela de Login.



PSIS

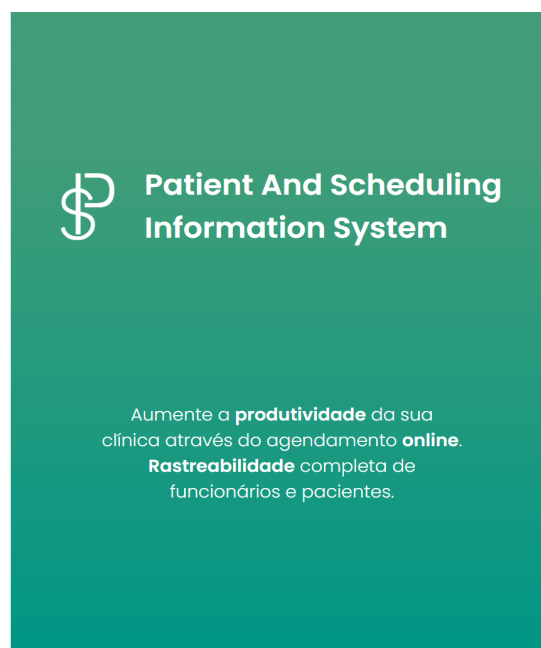
Acesse o painel da sua clínica

Clinica \*

Usuário \*

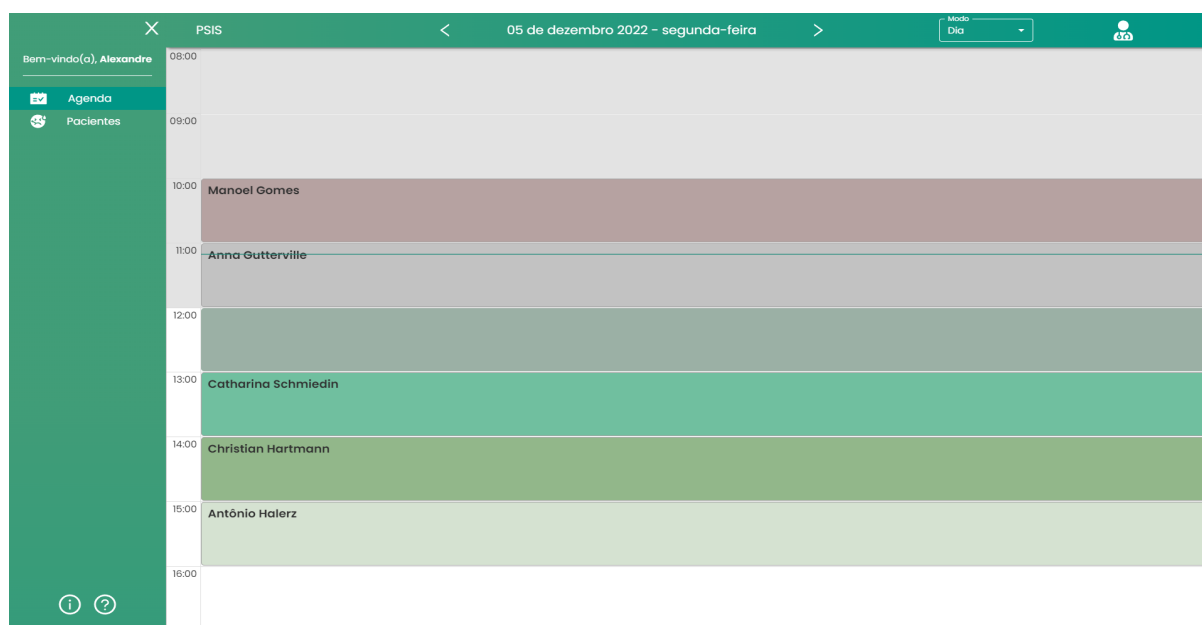
Senha \* 

ENTRAR



Fonte: Os autores.

FIGURA 8 - Tela da agenda.



PSIS 05 de dezembro 2022 - segunda-feira Modo Dia

Bem-vindo(a), Alexandre

Agenda Pacientes

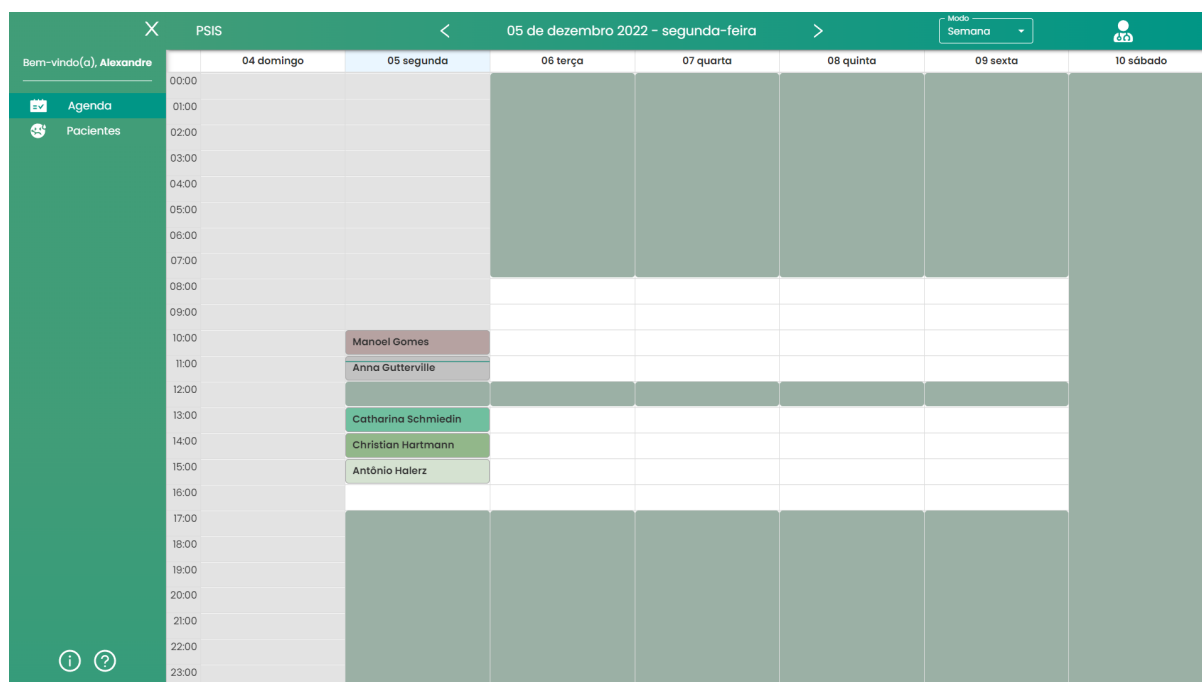
Horário	Paciente
08:00	
09:00	
10:00	Manoel Gomes
11:00	Anna Guterville
12:00	
13:00	Catharina Schmiedin
14:00	Christian Hartmann
15:00	Antônio Halerz
16:00	

1 2

Fonte: Os autores.

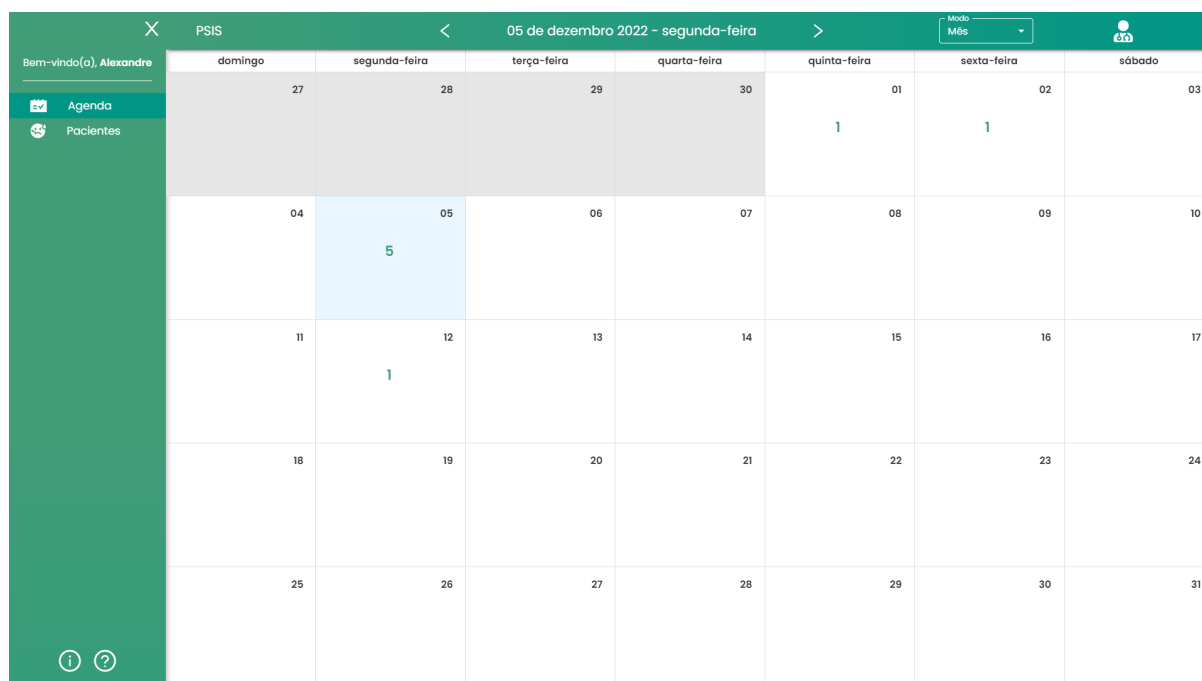


FIGURA 9 - Tela da agenda em modo semanal.

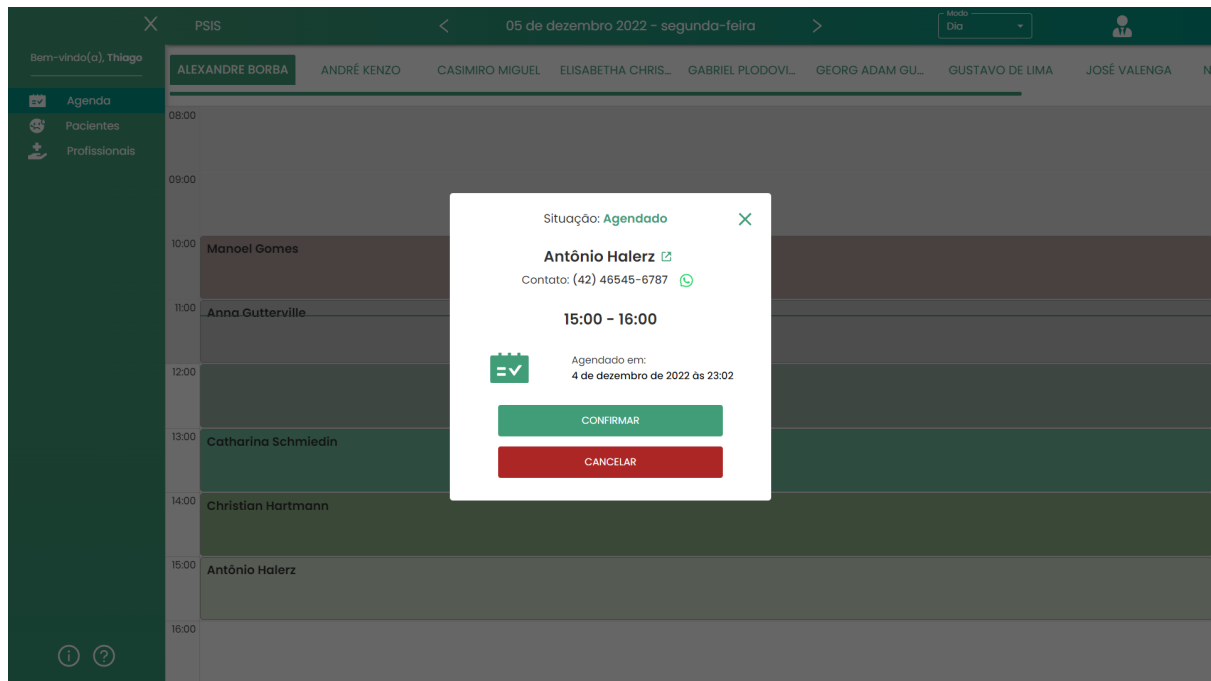


Fonte: Os autores.

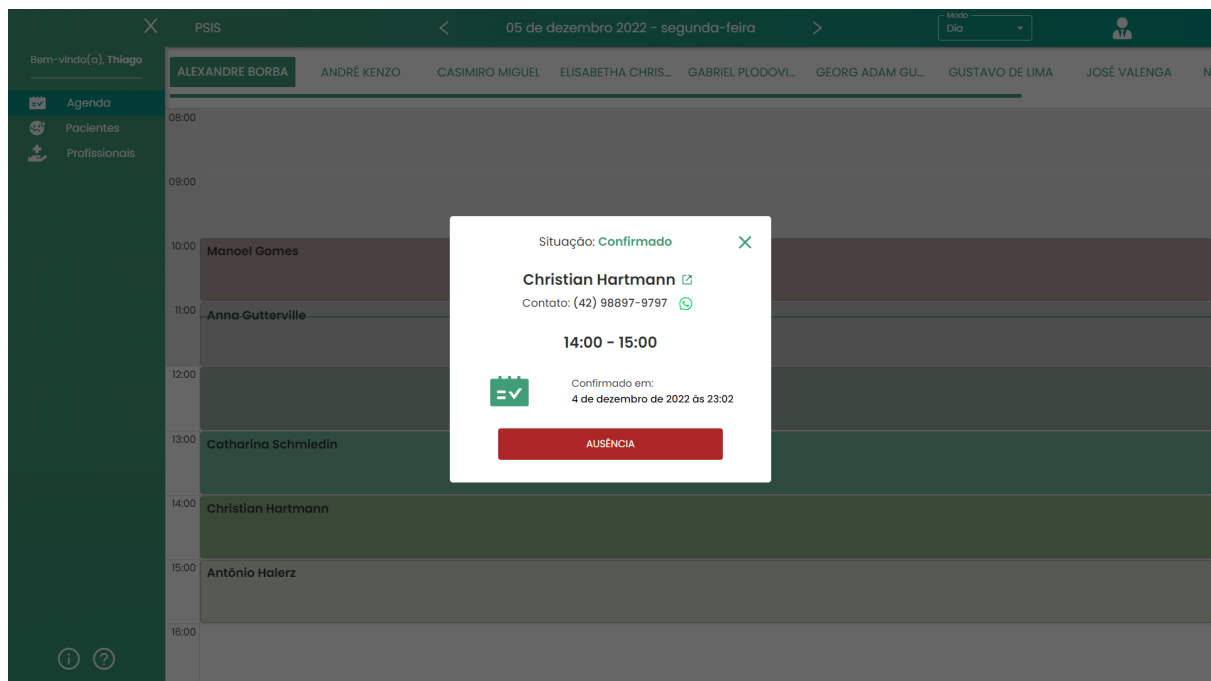
FIGURA 10 - Tela da agenda em modo mensal.



Fonte: Os autores.

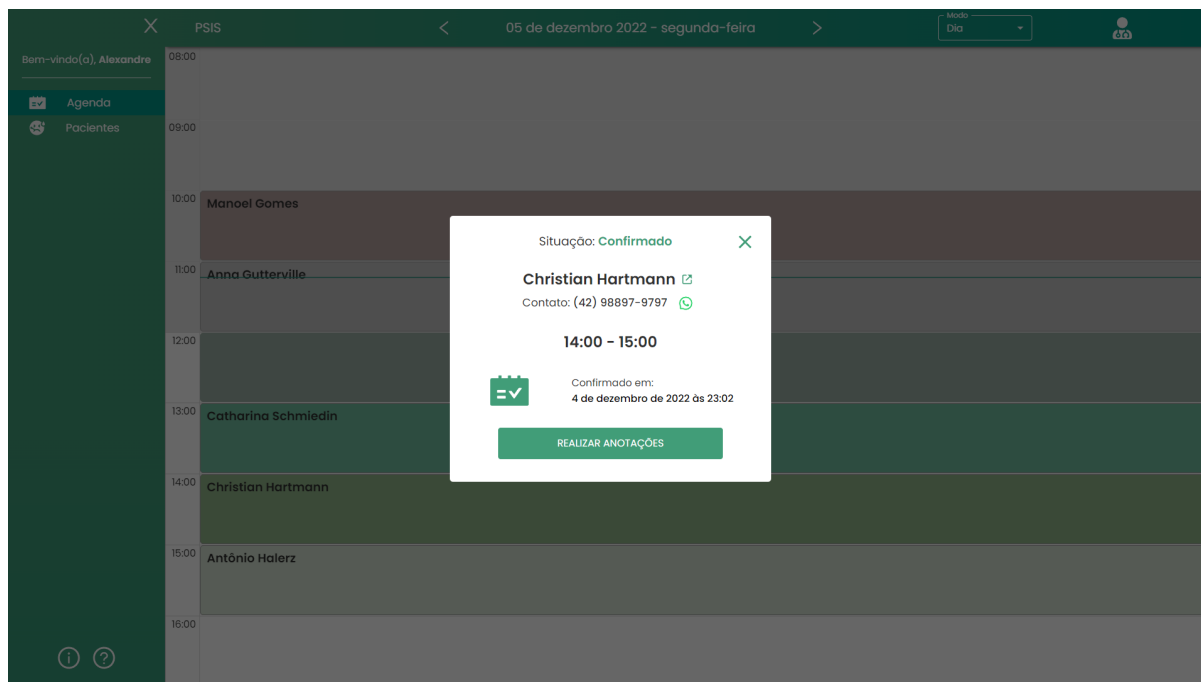
FIGURA 11 - Tela de visualização de uma consulta com *status* agendado.

Fonte: Os autores.

FIGURA 12 - Tela de visualização de uma consulta com *status* confirmado na visão do funcionário.

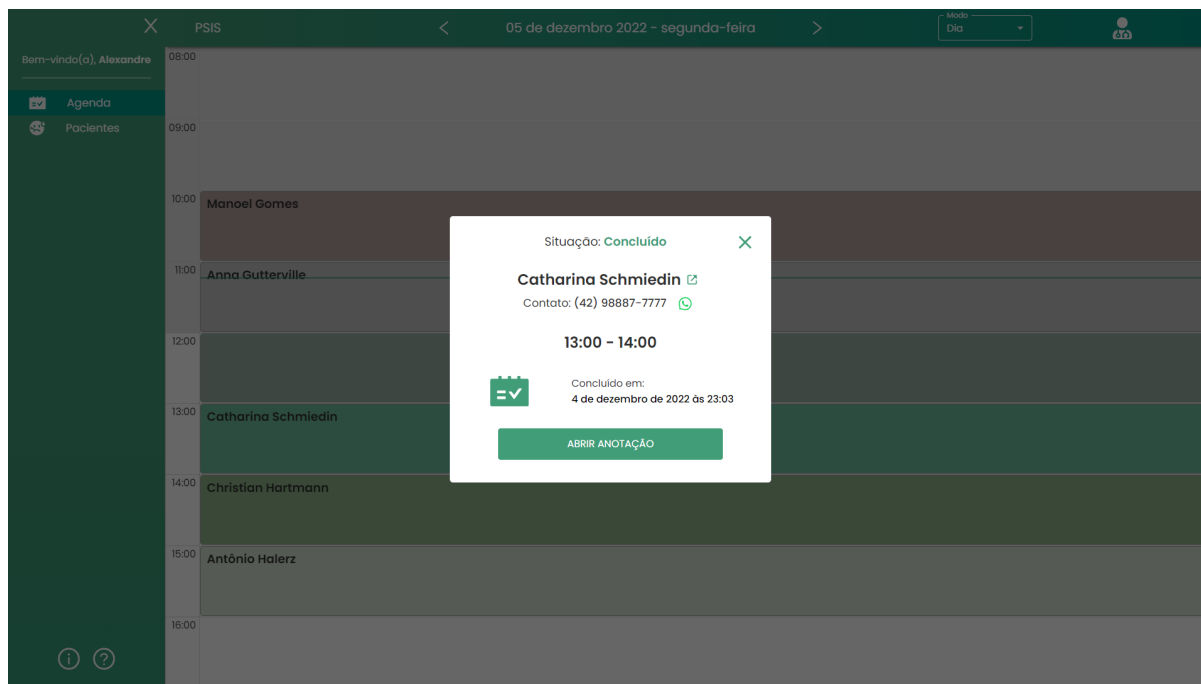
Fonte: Os autores.

FIGURA 13 - Tela de visualização de uma consulta com *status* confirmado na visão do profissional.



Fonte: Os autores.

FIGURA 14 - Tela de visualização de uma consulta com *status* concluído.



Fonte: Os autores.

FIGURA 15 - Tela de escolha de um espaço vazio para agendamento.

The screenshot shows a medical scheduling interface. At the top, there's a header with 'PSIS', navigation arrows, the date '05 de dezembro 2022 - segunda-feira', and a 'Modo Dia' dropdown. Below the header is a list of professionals: ALEXANDRE BORBA, ANDRÉ KENZO, CASIMIRO MIGUEL, ELISABETHA CHRIS..., GABRIEL PLODOVL..., GEORG ADAM GU..., GUSTAVO DE LIMA, JOSÉ VALENGA, and N. The left sidebar has a 'Bem-vindo(a), Thiago' greeting and three menu items: 'Agenda' (selected), 'Pacientes', and 'Profissionais'. The main area is a calendar grid for December 5, 2022, with time slots from 08:00 to 16:00. A modal window is open in the center, titled '5 de dezembro de 2022 | 16:00 - 17:00'. It has a 'Paciente' section with a 'Nome' dropdown menu. Below this is a green 'AGENDAR' button. The background calendar shows appointments for 'Manoel Gomes' at 10:00, 'Anna Gutierrez' at 11:00, 'Catharina Schmiedin' at 13:00, 'Christian Hartmann' at 14:00, and 'Antônio Halerz' at 15:00.

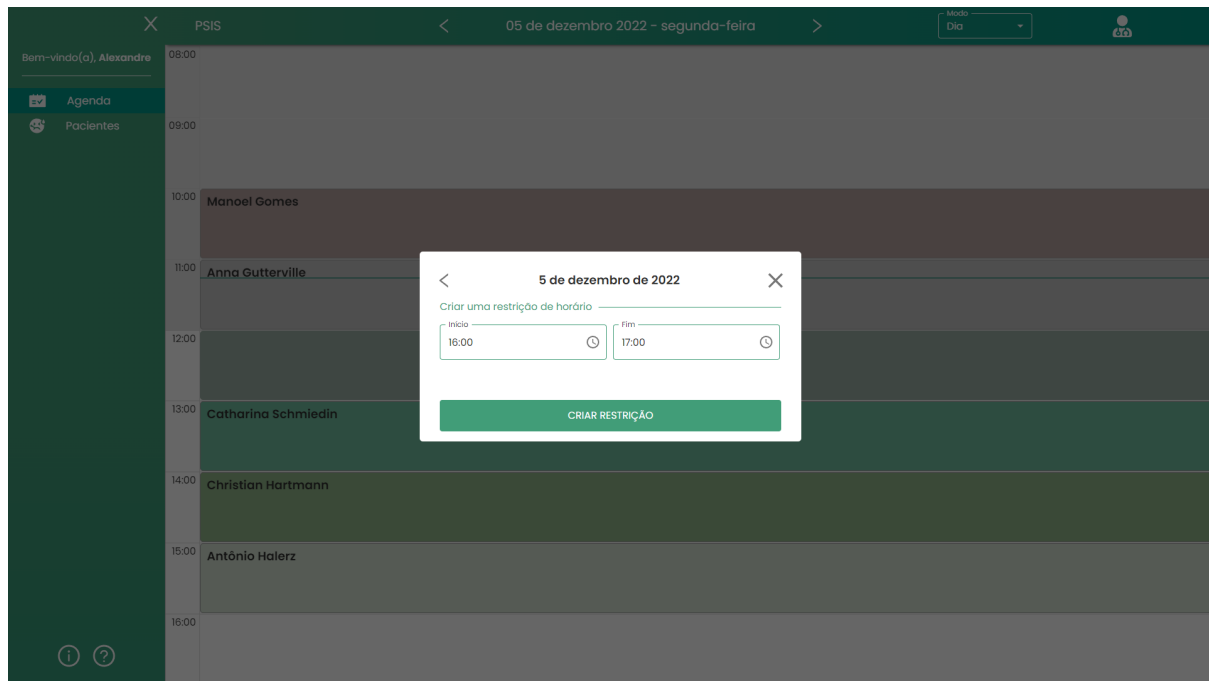
Fonte: Os autores.

FIGURA 16 - Tela de agendamento com dados do paciente preenchidos.

This screenshot shows the same medical scheduling interface as Figure 15, but with the patient data form in the modal window filled out. The modal window title is '5 de dezembro de 2022 | 16:00 - 17:00'. The 'Paciente' section now includes a 'Nome' dropdown menu with 'Anna Gutierrez' selected. Below this, there are two input fields: 'CPF' with the value '829.108.740-79' and 'Telefone' with the value '(42) 98898-7877'. The 'Responsável' section has a 'Nome' dropdown menu with 'Adão Gutierrez' selected, and two input fields: 'CPF' with the value '829.108.740-79' and 'Telefone' with the value '(42) 98898-7877'. The green 'AGENDAR' button is still present at the bottom of the modal. The background calendar and professional list remain the same as in Figure 15.

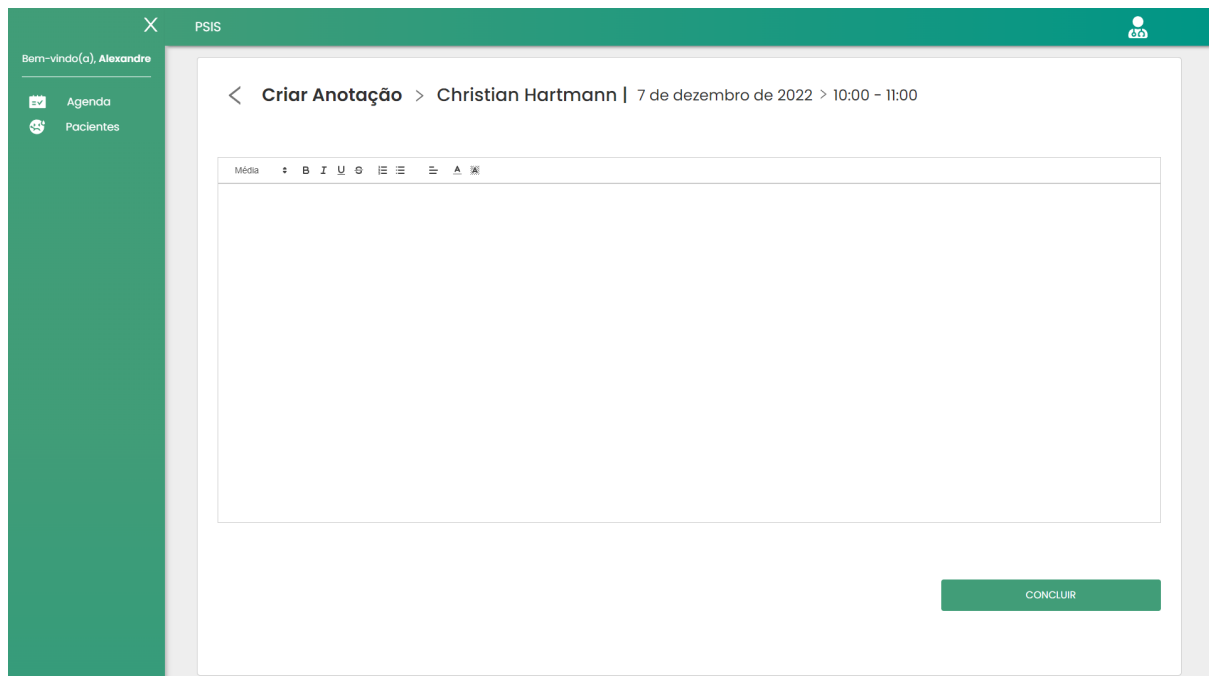
Fonte: Os autores.

FIGURA 17 - Tela de criação de restrição de horário na agenda.



Fonte: Os autores.

FIGURA 18 - Tela de criação de anotação após conclusão da consulta.



Fonte: Os autores.

FIGURA 19 - Tela de listagem de pacientes.

**Lista de Pacientes**

Nome, CPF ou e-mail

ADICIONAR

BUSCAR

Nome	CPF	Data de nascimento	Telefone	Ações
Anna Guterville	829.108.740-79	24/07/2010 (12 anos)	(42) 98899-7877	
Anna Müller	300.300.870-66	28/07/1999 (23 anos)	(42) 65778-9543	
Antônio Halerz	601.887.090-88	12/04/1988 (34 anos)	(42) 46545-6787	

**Dados pessoais**

Nome: Antônio Halerz  
CPF: 601.887.090-88  
Data de nascimento: 12/04/1988

Email:   
Estado civil: Viúvo(a)  
Gênero: Masculino

**Dados auxiliares**

Cidade: Ponta Grossa  
Estado: PR  
CEP: 84020-040

Bairro: Neves  
Logradouro: Rua Abílio Holzmann  
Telefone: (42) 46545-6787

Catharina Schmiedlin	644.558.410-16	13/10/2011 (11 anos)	(42) 98887-7777	
Christian Hartmann	021.535.480-04	13/10/2010 (12 anos)	(42) 98897-9797	
Danilo Reginaldo	127.553.890-87	27/04/1990 (32 anos)	(42) 98887-7777	
Jodo Specht	746.018.440-37	11/07/1994 (28 anos)	(42) 56756-7666	

1-10 de 18

Fonte: Os autores.

FIGURA 20 - Tela de cadastro de paciente.

**Criar Paciente**

**Dados Pessoais**

Nome \* Email

CPF \* Data de nascimento \* Estado civil \* Gênero \*

**Dados Auxiliares**

CEP Cidade Logradouro

Estado Bairro Telefone \*

☐ Paciente precisa de responsável

Campos marcados com \* são obrigatórios

SALVAR CANCELAR

Fonte: Os autores.

FIGURA 21 - Tela de listagem de consultas concluídas de um paciente.

**< Consultas Concluídas > Manoel Gomes**

Data inicial  Data final  **BUSCAR**

Data da consulta: 11/11/2022   10:00 - 11:00	Agendada em: 04/11/2022	Concluída em: 10/11/2022	→
Data da consulta: 04/11/2022   10:00 - 11:00	Agendada em: 04/11/2022	Concluída em: 10/11/2022	→
Data da consulta: 20/10/2022   11:00 - 12:00	Agendada em: 11/10/2022	Concluída em: 20/10/2022	→
Data da consulta: 19/10/2022   14:00 - 15:00	Agendada em: 15/10/2022	Concluída em: 19/10/2022	→
Data da consulta: 18/10/2022   11:00 - 12:00	Agendada em: 14/10/2022	Concluída em: 18/10/2022	→

1-5 de 5 < >

Fonte: Os autores.

FIGURA 22 - Tela de configurações de agenda do profissional.

**< Horários do Profissional > Alexandre**

Dias da semana ⓘ

**Domingo** Segunda-feira Terça-feira Quarta-feira Quinta-feira Sexta-feira Sábado

**Início e fim do expediente** | Duração base das consultas: 60 minutos. ⓘ

Início: 08:00  Fim: 17:00  ☐ Dia da semana sem expediente

**Intervalos - 8 restantes** ⓘ

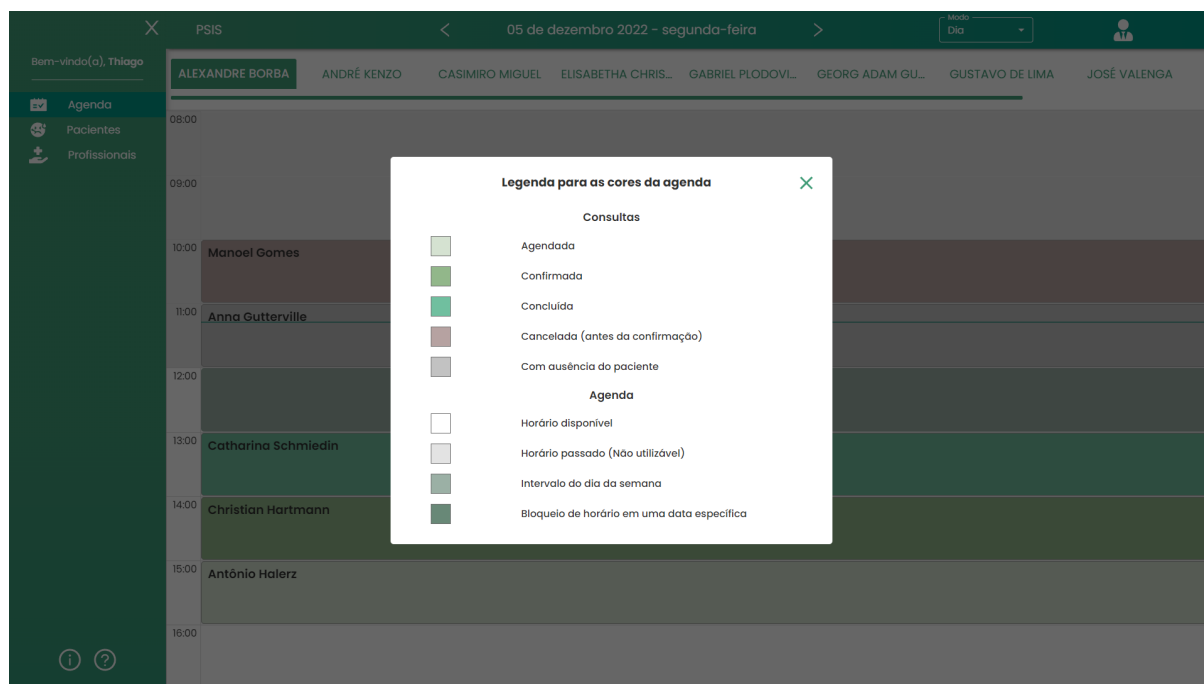
Início: 12:00 Fim: 13:00

**SALVAR**

Fonte: Os autores.

As Figuras 23 à 25 apresentam os resultados das interfaces principais em relação às ajudas fornecidas ao usuário, ou seja, informações pertinentes que foram inseridas em contextos específicos a fim de facilitar o uso do sistema por parte do usuário.

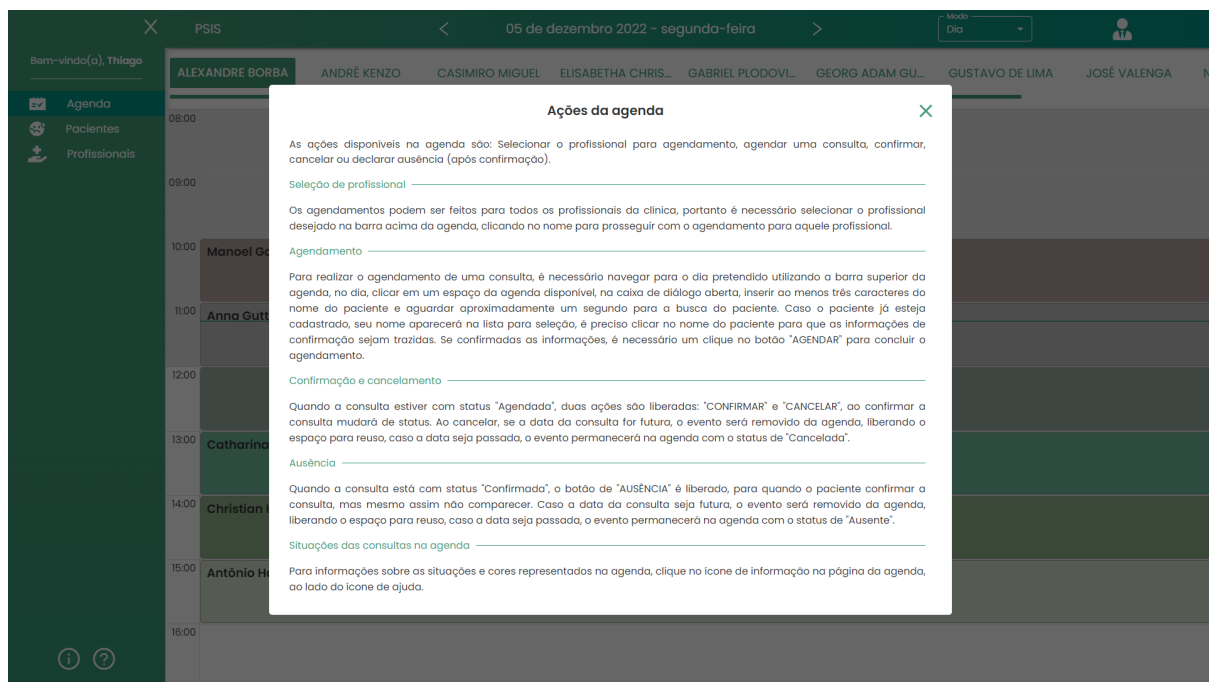
FIGURA 23 - Tela da agenda com a ajuda referente às cores dos eventos.



Fonte: Os autores.

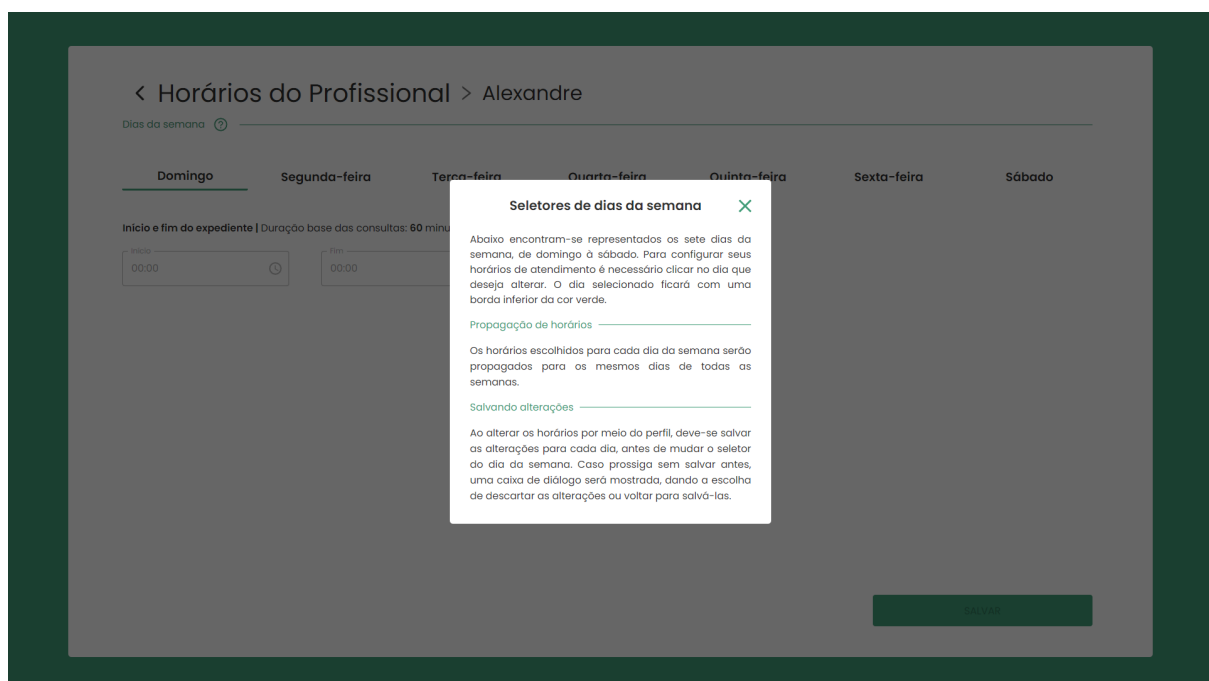


FIGURA 24 - Tela da agenda com a ajuda referente ao uso para colaboradores.



Fonte: Os autores.

FIGURA 25 - Tela de configurações de agenda do profissional com a ajuda aberta.



Fonte: Os autores.

## 7 PLANO DE TESTES

O plano de teste para o projeto foi criado em quadros com informações como nome, pré-condições - contendo os requisitos básicos para execução do teste, passos - descrevendo o roteiro a ser seguido - e resultado esperado.

O Quadro 25 possui as informações dos principais testes para as operações com a entidade *Consulta*. O Quadro 26 apresenta os testes para as principais operações referentes à entidade *Paciente*. O Quadro 27, por fim, descreve os testes relacionados à agenda dos profissionais.

QUADRO 25 - Descrição do plano de testes para o módulo referentes às consultas.

(continua)

Nome	Pré-condições	Passos	Resultado esperado
Cadastrar consulta como funcionário	O profissional deve estar configurado e possuir a agenda semanal de segunda-feira das 08:00 às 17:00, e o paciente "Manoel Gomes" deve estar cadastrado.	1) Acessar o item "Agenda" no menu lateral. 2) Selecionar o profissional no seletor localizado acima do calendário. 3) Encontrar uma segunda-feira futura. 4) Selecionar o horário das 15:00 às 16:00. 5) No modal de escolha, informar o nome "Manoel" e selecionar o paciente correto. 5) Clicar em "Agendar".	Adição de um novo evento no calendário no horário das 15:00 às 16:00.
Cancelar consulta	A consulta deve ter sido previamente cadastrada.	1) Acessar o item "Agenda" no menu lateral. 2) Selecionar o profissional no seletor localizado acima do calendário. 3) Procurar e selecionar a consulta para cancelar. 4) Clicar no botão "Cancelar".	Alteração da cor do evento no calendário para datas passadas, ou remoção do evento para datas futuras.
Confirmar consulta	A consulta deve ter sido previamente cadastrada - deve estar com a situação "Agendada".	1) Acessar o item "Agenda" no menu lateral. 2) Selecionar o profissional no seletor localizado acima do calendário. 3) Procurar e selecionar a consulta para cancelar. 3) Clicar no botão "Confirmar".	Alteração da cor do evento no calendário.

Fonte: Os autores.

QUADRO 25 - Descrição do plano de testes para o módulo referentes às consultas.

(conclusão)

Nome	Pré-condições	Passos	Resultado esperado
Concluir consulta	A consulta deve ter sido previamente cadastrada e deve estar com a situação "Confirmada". Apenas o profissional pode realizar esta operação.	1) Acessar o item "Agenda" no menu lateral. 2) Selecionar a consulta para concluir. 3) Clicar no botão "Realizar Anotações". 4) Fazer as anotações, se necessário. 5) Clicar no botão "Concluir".	Abertura do modal com a mensagem "Operação realizada com sucesso." e mudança na cor do evento na agenda.

Fonte: Os autores.

QUADRO 26 - Descrição do plano de testes para o módulo referentes às operações básicas sobre a entidade *Paciente*.

(continua)

Nome	Pré-condições	Passos	Resultado esperado
Cadastrar paciente sem responsável	Estar autenticado em uma conta de funcionário. Os dados do paciente devem conter uma data de nascimento menor que a data atual, nome e, se preenchidos, CPF e e-mail únicos e válidos.	1) Acessar o item "Pacientes" no menu lateral. 2) Clicar em "Adicionar". 3) Preencher o nome do paciente com "Manoel Gomes". 4) Preencher o CPF do paciente com "466.583.510-15". 5) No campo "Data de nascimento", clicar no ícone de agenda, selecionar o ano "2000", o mês "agosto" e o dia "31". 6) Clicar no campo "Estado civil" e selecionar a opção "Solteiro(a)". 7) Clicar no campo "Gênero" e selecionar a opção "Masculino". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "Operação realizada com sucesso", campos do formulário limpos.

Fonte: Os autores.

QUADRO 26 - Descrição do plano de testes para o módulo referentes às operações básicas sobre a entidade *Paciente*.

(conclusão)

Nome	Pré-condições	Passos	Resultado esperado
Cadastrar paciente sem responsável com CPF já cadastrado	Estar autenticado em uma conta de funcionário. O CPF preenchido para cadastrar o paciente já deve ter sido cadastrado por outro. Cadastrar um paciente com o CPF "466.583.510-15" previamente.	1) Acessar o item "Pacientes" no menu lateral. 2) Clicar em "Adicionar". 3) Preencher o nome do paciente com "Manoel Gomes". 4) Preencher o CPF do paciente com "466.583.510-15". 5) No campo "Data de nascimento", clicar no ícone de agenda, selecionar o ano "2000", o mês "agosto" e o dia "31". 6) Clicar no campo "Estado civil" e selecionar a opção "Solteiro(a)". 7) Clicar no campo "Gênero" e selecionar a opção "Masculino". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "O CPF informado já está cadastrado."
Editar Paciente	Estar autenticado em uma conta de funcionário. O paciente deve constar na tabela de pacientes.	1) Acessar o item "Pacientes" no menu lateral. 2) Encontrar o paciente de nome "Manoel Gomes". 3) Clicar no ícone de edição na coluna de ações. 4) No campo "email", adicionar "mgomes@gmail.com". 5) No campo "telefone", adicionar "(42) 99999-9999". 6) Clicar no botão "Salvar".	Abertura do modal com a mensagem "Operação realizada com sucesso", e redirecionamento para a tabela de pacientes.
Deletar Paciente	Estar autenticado em uma conta de funcionário. O paciente deve constar na tabela de pacientes.	1) Acessar o item "Pacientes" no menu lateral. 2) Encontrar o paciente de nome "Manoel Gomes". 3) Clicar no ícone de deleção na coluna de ações. 4) Clicar em "Confirmar" no modal de confirmação.	Mensagem "Operação realizada com sucesso.", e atualização da tabela removendo o paciente deletado.

Fonte: Os autores.

QUADRO 27 - Descrição do plano de testes para o módulo referentes às operações sobre a agenda dos profissionais.

(continua)

Nome	Pré-condições	Passos	Resultado esperado
Alteração correta do horário semanal para segunda-feira.	O profissional deve constar na base de dados e deve estar ativo.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 5) Navegar para a seção "Início e fim do expediente". 6) Clicar no ícone de relógio no <i>input</i> "Início" e selecionar "08:00". 7) Clicar no ícone de relógio no <i>input</i> "Fim" e selecionar "17:00". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "Operação realizada com sucesso" e botão de salvar será desabilitado.
Alteração do horário semanal para segunda-feira: intervalo inválido.	O profissional deve constar na base de dados e deve estar ativo.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 5) Navegar para a seção "Início e fim do expediente". 6) Clicar no ícone de relógio no <i>input</i> "Início" e selecionar "12:00". 7) Clicar no ícone de relógio no <i>input</i> "Fim" e selecionar "08:00". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "O intervalo do horário é inválido, o horário de início deve ser inferior ao de término" e botão de salvar será desabilitado.
Alteração do horário semanal para segunda-feira: conflito com consulta não concluída em data futura.	O profissional deve constar na base de dados e deve estar ativo, o horário de expediente de segunda-feira deve ser das 08:00 às 17:00, e deve possuir uma consulta marcada em alguma segunda-feira de data futura no horário das 08:00 às 09:00.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 5) Navegar para a seção "Início e fim do expediente". 6) Clicar no ícone de relógio no <i>input</i> "Início" e selecionar "09:00". 7) Clicar em "Salvar".	Abertura do modal com a mensagem "Não é possível realizar a operação na agenda pois existem consultas pendentes em datas futuras para Segunda-feira." e botão de salvar será desabilitado.

Fonte: Os autores.

QUADRO 27 - Descrição do plano de testes para o módulo referentes às operações sobre a agenda dos profissionais.

(continuação)

Nome	Pré-condições	Passos	Resultado esperado
Criação de restrições no horário para alguma segunda-feira do calendário.	O profissional deve constar na base de dados e deve estar ativo, e a agenda semanal de segunda-feira deve compreender o horário das 08:00 às 17:00.	1) Acessar o item "Agenda" no menu lateral. 2) Encontrar qualquer segunda-feira no calendário. 3) Clicar no intervalo do calendário com início às 14:00. 4) Clicar no ícone de cadeado vermelho no canto superior esquerdo do modal. 5) Clicar no ícone de relógio no <i>input</i> "Fim" e selecionar "16:00". 6) Clicar no botão "Criar restrição".	Fechamento do modal e adição do evento na agenda no horário das 14:00 às 16:00.
Criação de restrições na agenda semanal de segunda-feira.	O profissional deve constar na base de dados e deve estar ativo, e a agenda do dia da semana deve possuir expediente (o <i>checkbox</i> "Dia da semana sem expediente" deve estar desmarcado).	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 4) Navegar para a seção "Intervalos". 6) Clicar no ícone de adição ao lado direito dos campos. 7) Clicar no ícone de relógio no <i>input</i> "Início" e selecionar "12:00". 8) Clicar no ícone de relógio no <i>input</i> "Fim" e selecionar "13:00". 7) Clicar em "Criar Intervalo". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "Operação realizada com sucesso" e botão de salvar será desabilitado.
Criação de restrições na agenda semanal de segunda-feira: restrição fora do intervalo de trabalho.	O profissional deve constar na base de dados e deve estar ativo, e a agenda de segunda-feira deve possuir expediente das 08:00 às 17:00.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 4) Navegar para a seção "Intervalos". 6) Clicar no ícone de adição ao lado direito dos campos. 7) Clicar no ícone de relógio no <i>input</i> "Início" e selecionar "18:00". 8) Clicar no ícone de relógio no <i>input</i> "Fim" e selecionar "19:00". 7) Clicar em "Criar Intervalo". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "Não é possível cadastrar o 1º bloqueio pois está fora da agenda de Segunda-feira." e botão de salvar será desabilitado.

Fonte: Os autores.

QUADRO 27 - Descrição do plano de testes para o módulo referentes às operações sobre a agenda dos profissionais.

(continuação)

Nome	Pré-condições	Passos	Resultado esperado
Criação de restrições na agenda semanal de segunda-feira: restrição fora do intervalo de trabalho.	O profissional deve constar na base de dados e deve estar ativo, e a agenda de segunda-feira deve possuir expediente das 08:00 às 17:00.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 4) Navegar para a seção "Intervalos". 6) Clicar no ícone de adição ao lado direito dos campos. 7) Clicar no ícone de relógio no <i>input</i> "Início" e selecionar "18:00". 8) Clicar no ícone de relógio no <i>input</i> "Fim" e selecionar "19:00". 7) Clicar em "Criar Intervalo". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "Não é possível cadastrar o 1º bloqueio pois está fora da agenda de Segunda-feira." e botão de salvar será desabilitado.
Criação de restrições na agenda semanal de segunda-feira: união com restrições gerais (por data).	O profissional deve constar na base de dados e deve estar ativo, a agenda de segunda-feira deve possuir expediente das 08:00 às 17:00 e deve possuir uma restrição em alguma segunda-feira futura das 14:00 às 15:00.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 4) Navegar para a seção "Intervalos". 6) Clicar no ícone de adição ao lado direito dos campos. 7) Clicar no ícone de relógio no <i>input</i> "Início" e selecionar "13:30". 8) Clicar no ícone de relógio no <i>input</i> "Fim" e selecionar "14:30". 7) Clicar em "Criar Intervalo". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "Operação realizada com sucesso" e botão de salvar será desabilitado. No calendário - no dia da restrição por data - os bloqueios irão se unir, ficando uma restrição de agenda das 13:30 às 14:30 e uma restrição de horário das 14:30 às 15:00.

Fonte: Os autores.

QUADRO 27 - Descrição do plano de testes para o módulo referentes às operações sobre a agenda dos profissionais.

(conclusão)

Nome	Pré-condições	Passos	Resultado esperado
Criação de restrições na agenda semanal de segunda-feira: conflito com consulta não concluída em data futura.	O profissional deve constar na base de dados e deve estar ativo, o horário de expediente de segunda-feira deve ser das 08:00 às 17:00, e deve possuir uma consulta marcada em alguma segunda-feira de data futura no horário das 14:00 às 15:00.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 4) Navegar para a seção "Intervalos". 6) Clicar no ícone de adição ao lado direito dos campos. 7) Clicar no ícone de relógio no <i>input</i> "Início" e selecionar "13:30". 8) Clicar no ícone de relógio no <i>input</i> "Fim" e selecionar "14:30". 7) Clicar em "Criar Intervalo". 8) Clicar em "Salvar".	Abertura do modal com a mensagem "Não é possível cadastrar o 1º bloqueio pois existem consultas pendentes em datas futuras para Segunda-feira." e botão de salvar será desabilitado.
Remover segunda-feira da agenda semana de trabalho.	O profissional deve constar na base de dados e deve estar ativo, e deve possuir a agenda de segunda-feira configurada com qualquer intervalo.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 5) Selecionar o <i>checkbox</i> "Dia da semana sem expediente" ao lado direito dos <i>inputs</i> da seção "Início e fim do expediente". 6) Clicar em "Salvar".	Abertura do modal com a mensagem "Operação realizada com sucesso." e botão de salvar será desabilitado. Na agenda, todas as segunda-feiras terão a mensagem "Este dia não possui expediente".
Remover segunda-feira da agenda semana de trabalho: conflito com consultas futuras não concluídas.	O profissional deve constar na base de dados e deve estar ativo, deve possuir a agenda de segunda-feira configurada com qualquer intervalo e uma consulta não concluída para segunda-feira.	1) Clicar no botão de perfil na barra superior. 2) Clicar em "Perfil". 3) Clicar no ícone de relógio no campo superior direito. 4) Selecionar "Segunda-feira" na aba de dias da semana. 5) Selecionar o <i>checkbox</i> "Dia da semana sem expediente" ao lado direito dos <i>inputs</i> da seção "Início e fim do expediente". 6) Clicar em "Salvar".	Abertura do modal com a mensagem "Não é possível realizar a operação na agenda pois existem consultas pendentes em datas futuras para Segunda-feira." e botão de salvar será desabilitado.

Fonte: Os autores.



## 8 SCRIPT PARA CRIAÇÃO DO BANCO DE DADOS

```
CREATE SEQUENCE gerar_codigo_base_clinica INCREMENT 1000 START 1000;
```

```
CREATE TABLE "clinica" (  
  "id" UUID NOT NULL,  
  "email" VARCHAR(100) NOT NULL,  
  "nome" VARCHAR(100) NOT NULL,  
  "codigo_base" INTEGER NOT NULL DEFAULT NEXTVAL('gerar_codigo_base_clinica'),  
  CONSTRAINT "clinica_pkey" PRIMARY KEY ("id")  
);
```

```
CREATE TABLE "pessoa" (  
  "id" UUID NOT NULL,  
  "email" VARCHAR(100),  
  "nome" VARCHAR(100) NOT NULL,  
  "classe_dominio" VARCHAR(32) NOT NULL,  
  "cpf" VARCHAR(11),  
  "data_nascimento" DATE NOT NULL,  
  "telefone" VARCHAR(11) NOT NULL,  
  "ativo" BOOLEAN NOT NULL DEFAULT true,  
  "data_criacao" TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  "id_clinica" UUID NOT NULL,  
  CONSTRAINT "pessoa_pkey" PRIMARY KEY ("id")  
);
```

```
CREATE TABLE "endereco" (  
  "id" UUID NOT NULL,  
  "cidade" VARCHAR(100) NOT NULL,  
  "bairro" VARCHAR(100),  
  "estado" VARCHAR(100) NOT NULL,  
  "logradouro" VARCHAR(255) NOT NULL,  
  "CEP" VARCHAR(8) NOT NULL,  
  "id_pessoa" UUID NOT NULL,  
  CONSTRAINT "endereco_pkey" PRIMARY KEY ("id")  
);
```

```
CREATE TABLE "usuario" (  
  "nome_usuario" VARCHAR(100) NOT NULL,  
  "bloqueado" BOOLEAN NOT NULL DEFAULT false,  
  "tentativas" INTEGER NOT NULL DEFAULT 0,  
  "senha" VARCHAR(255) NOT NULL,  
  "id" UUID NOT NULL,  
  "id_papel_usuario" INTEGER NOT NULL,  
  "data_ultimo_login_errado" TIMESTAMP,  
  CONSTRAINT "usuario_pkey" PRIMARY KEY ("id")  
);
```

```
CREATE TABLE "papel_usuario" (
  "id" SERIAL NOT NULL,
  "nome" VARCHAR(36) NOT NULL,
  CONSTRAINT "papel_usuario_pkey" PRIMARY KEY ("id")
);
```

```
CREATE TABLE "permissoes" (
  "id" SERIAL NOT NULL,
  "nome" VARCHAR(36) NOT NULL,
  "id_papel_usuario" INTEGER NOT NULL,
  CONSTRAINT "permissoes_pkey" PRIMARY KEY ("id")
);
```

```
CREATE TABLE "paciente" (
  "genero" SMALLINT NOT NULL,
  "estado_civil" SMALLINT NOT NULL,
  "id" UUID NOT NULL,
  CONSTRAINT "paciente_pkey" PRIMARY KEY ("id")
);
```

```
CREATE TABLE "profissional" (
  "profissao" VARCHAR(255) NOT NULL,
  "especializacao" VARCHAR(255),
  "duracao_base" INTEGER NOT NULL DEFAULT 60,
  "registro" VARCHAR(255) NOT NULL,
  "id" UUID NOT NULL,
  CONSTRAINT "profissional_pkey" PRIMARY KEY ("id")
);
```

```
CREATE TABLE "responsavel" (
  "id_pessoa" UUID NOT NULL,
  "id_paciente" UUID NOT NULL,
  CONSTRAINT "responsavel_pkey" PRIMARY KEY ("id_paciente")
);
```

```
CREATE TABLE "agenda_semanal" (
  "id" UUID NOT NULL,
  "horario_inicio" TIME NOT NULL,
  "horario_fim" TIME NOT NULL,
  "dia_semana" SMALLINT NOT NULL,
  "id_profissional" UUID NOT NULL,
  CONSTRAINT "agenda_semanal_pkey" PRIMARY KEY ("id")
);
```

```
CREATE TABLE "restricoes_agenda_semanal" (
  "id" UUID NOT NULL,
  "horario_inicio" TIME NOT NULL,
  "horario_fim" TIME NOT NULL,
  "id_agenda_semanal" UUID NOT NULL,
  CONSTRAINT "restricoes_agenda_semanal_pkey" PRIMARY KEY ("id")
);
```

```
CREATE TABLE "restricoes_horario" (
  "id" UUID NOT NULL,
  "horario_inicio" TIME NOT NULL,
  "horario_fim" TIME NOT NULL,
  "data" DATE NOT NULL,
  "id_profissional" UUID NOT NULL,
  CONSTRAINT "restricoes_horario_pkey" PRIMARY KEY ("id")
);
```

```
CREATE TABLE "consulta" (
  "id" UUID NOT NULL,
  "situacao" SMALLINT NOT NULL,
  "data_agendamento" TIMESTAMP NOT NULL,
  "data_ultima_atualizacao" TIMESTAMP NOT NULL,
  "data_criacao" TIMESTAMP NOT NULL,
  "observacoes" TEXT,
  "id_paciente" UUID NOT NULL,
  "id_profissional" UUID NOT NULL,
  "id_funcionario" UUID NOT NULL,
  CONSTRAINT "consulta_pkey" PRIMARY KEY ("id")
);
```

```
CREATE UNIQUE INDEX "clinica_email_key" ON "clinica"("email");
CREATE UNIQUE INDEX "endereco_id_pessoa_key" ON "endereco"("id_pessoa");
CREATE UNIQUE INDEX "usuario_id_key" ON "usuario"("id");
CREATE UNIQUE INDEX "paciente_id_key" ON "paciente"("id");
CREATE UNIQUE INDEX "profissional_id_key" ON "profissional"("id");
```

```
ALTER TABLE "pessoa" ADD CONSTRAINT "pessoa_id_clinica_fkey" FOREIGN KEY
("id_clinica") REFERENCES "clinica"("id") ON DELETE RESTRICT ON UPDATE
CASCADE;
```

```
ALTER TABLE "endereco" ADD CONSTRAINT "endereco_id_pessoa_fkey" FOREIGN KEY
("id_pessoa") REFERENCES "pessoa"("id") ON DELETE RESTRICT ON UPDATE
CASCADE;
```

```
ALTER TABLE "usuario" ADD CONSTRAINT "usuario_id_fkey" FOREIGN KEY ("id")
REFERENCES "pessoa"("id") ON DELETE RESTRICT ON UPDATE CASCADE;
```

```
ALTER TABLE "usuario" ADD CONSTRAINT "usuario_id_pepel_usuario_fkey" FOREIGN
KEY ("id_papel_usuario") REFERENCES "papel_usuario"("id") ON DELETE RESTRICT ON
UPDATE CASCADE;
```

```
ALTER TABLE "permissoes" ADD CONSTRAINT "permissoes_id_papel_usuario_fkey"
FOREIGN KEY ("id_papel_usuario") REFERENCES "papel_usuario"("id") ON DELETE
RESTRICT ON UPDATE CASCADE;
```

```
ALTER TABLE "paciente" ADD CONSTRAINT "paciente_id_fkey" FOREIGN KEY ("id")
REFERENCES "pessoa"("id") ON DELETE RESTRICT ON UPDATE CASCADE;
```

```
ALTER TABLE "profissional" ADD CONSTRAINT "profissional_id_fkey" FOREIGN KEY ("id")
REFERENCES "usuario"("id") ON DELETE RESTRICT ON UPDATE CASCADE;
```

```
ALTER TABLE "responsavel" ADD CONSTRAINT "responsavel_id_pessoa_fkey" FOREIGN
KEY ("id_pessoa") REFERENCES "pessoa"("id") ON DELETE RESTRICT ON UPDATE
CASCADE;
```

```
ALTER TABLE "responsavel" ADD CONSTRAINT "responsavel_id_paciente_fkey"
FOREIGN KEY ("id_paciente") REFERENCES "paciente"("id") ON DELETE RESTRICT ON
UPDATE CASCADE;
```

```
ALTER TABLE "agenda_semanal" ADD CONSTRAINT
"agenda_semanal_id_profissional_fkey" FOREIGN KEY ("id_profissional") REFERENCES
"profissional"("id") ON DELETE RESTRICT ON UPDATE CASCADE;
```

```
ALTER TABLE "restricoes_agenda_semanal" ADD CONSTRAINT
"restricoes_agenda_semanal_id_agenda_semanal_fkey" FOREIGN KEY
("id_agenda_semanal") REFERENCES "agenda_semanal"("id") ON DELETE RESTRICT
ON UPDATE CASCADE;
```

```
ALTER TABLE "restricoes_horario" ADD CONSTRAINT
"restricoes_horario_id_profissional_fkey" FOREIGN KEY ("id_profissional") REFERENCES
"profissional"("id") ON DELETE RESTRICT ON UPDATE CASCADE;
```

```
ALTER TABLE "consulta" ADD CONSTRAINT "consulta_id_funcionario_fkey" FOREIGN
KEY ("id_funcionario") REFERENCES "usuario"("id") ON DELETE RESTRICT ON UPDATE
CASCADE;
```

```
ALTER TABLE "consulta" ADD CONSTRAINT "consulta_id_paciente_fkey" FOREIGN KEY
("id_paciente") REFERENCES "paciente"("id") ON DELETE RESTRICT ON UPDATE
CASCADE;
```

```
ALTER TABLE "consulta" ADD CONSTRAINT "consulta_id_profissional_fkey" FOREIGN
KEY ("id_profissional") REFERENCES "profissional"("id") ON DELETE RESTRICT ON
UPDATE CASCADE;
```

```

CREATE FUNCTION previnir_consultas_mesmo_horario() RETURNS TRIGGER
AS $$
    DECLARE
        DURACAO_CONSULTA INTEGER;
        ID_CONSULTA_EXISTENTE CHARACTER VARYING(36) := NULL;
    BEGIN
        SELECT p.duracao_base INTO DURACAO_CONSULTA
        FROM profissional p
        WHERE p.id = NEW.id_profissional;

        SELECT c.id INTO ID_CONSULTA_EXISTENTE
        FROM consulta c
        WHERE
            AND c.id_profissional = NEW.id_profissional
            AND c.situacao <> 4
            AND (
                (NEW.data_agendamento = c.data_agendamento)
                OR (
                    NEW.data_agendamento > c.data_agendamento
                    AND NEW.data_agendamento < (c.data_agendamento + (DURACAO_CONSULTA*
interval '1 minute'))
                )
                OR (
                    NEW.data_agendamento < c.data_agendamento
                    AND (NEW.data_agendamento + (DURACAO_CONSULTA* interval '1 minute')) >
c.data_agendamento
                )
            )
        LIMIT 1;

        IF (coalesce(ID_CONSULTA_EXISTENTE, '') <> '') THEN
            raise exception 'TRIGGER_RAISE_EXCEPTION_0';
        END IF;
        RETURN NEW;
    END;
$$
LANGUAGE plpgsql;

CREATE TRIGGER trg_previnir_consultas_mesmo_horario
BEFORE INSERT ON "consulta"
FOR EACH ROW
EXECUTE PROCEDURE previnir_consultas_mesmo_horario();

```

```

CREATE FUNCTION bloquear_consulta_fora_da_agenda_semanal() RETURNS TRIGGER
AS $$
    DECLARE
        ID_AGENDA_SEMANAL_EXISTENTE CHARACTER VARYING(36) := NULL;
        ID_RESTRICAO_AGENDA_SEMANAL_EXISTENTE CHARACTER VARYING(36) :=
NULL;
    BEGIN
        SELECT ag.id INTO ID_AGENDA_SEMANAL_EXISTENTE
        FROM agenda_semanal ag
        WHERE
            ag.id_profissional = NEW.id_profissional
            AND ag.dia_semana = ((extract(DOW FROM NEW.data_agendamento::TIMESTAMP)) +
1)
            AND (NEW.data_agendamento::TIME) < ag.horario_fim
            AND (NEW.data_agendamento::TIME) > ag.horario_inicio;

        IF (coalesce(ID_AGENDA_SEMANAL_EXISTENTE, '') = '') THEN
            raise exception 'TRIGGER_RAISE_EXCEPTION_1';
        END IF;

        SELECT rag.id INTO ID_RESTRICAO_AGENDA_SEMANAL_EXISTENTE
        FROM restricoes_agenda_semanal rag
        WHERE
            rag.id_agenda_semanal = (ID_AGENDA_SEMANAL_EXISTENTE::UUID)
            AND (NEW.data_agendamento::TIME) < rag.horario_fim
            AND (NEW.data_agendamento::TIME) > rag.horario_inicio;

        IF (coalesce(ID_RESTRICAO_AGENDA_SEMANAL_EXISTENTE, '') <> '') THEN
            raise exception 'TRIGGER_RAISE_EXCEPTION_2';
        END IF;

        RETURN NEW;
    END;
$$
LANGUAGE plpgsql;

CREATE TRIGGER trg_bloquear_consulta_fora_da_agenda_semanal
BEFORE INSERT ON "consulta"
FOR EACH ROW
EXECUTE PROCEDURE bloquear_consulta_fora_da_agenda_semanal();

```

```

CREATE FUNCTION verificar_restricoes_horario_profissional() RETURNS TRIGGER
AS $$
    DECLARE
        ID_RESTRICAO_HORARIO_EXISTENTE CHARACTER VARYING(36) := NULL;
    BEGIN
        SELECT r.id INTO ID_RESTRICAO_HORARIO_EXISTENTE
        FROM restricoes_horario r
        WHERE
            r.id_profissional = NEW.id_profissional
            AND r.data = (NEW.data_agendamento::DATE)
            AND (NEW.data_agendamento::TIME) < r.horario_fim
            AND (NEW.data_agendamento::TIME) > r.horario_inicio;

        IF (coalesce(ID_RESTRICAO_HORARIO_EXISTENTE, '') <> '') THEN
            raise exception 'TRIGGER_RAISE_EXCEPTION_3';
        END IF;

        RETURN NEW;
    END;
$$
LANGUAGE plpgsql;

```

```

CREATE TRIGGER trg_verificar_restricoes_horario_profissional
BEFORE INSERT ON "consulta"
FOR EACH ROW
EXECUTE PROCEDURE verificar_restricoes_horario_profissional();

```

```

ALTER TABLE "pessoa"
ADD CONSTRAINT pessoa_check_nascimento
CHECK ("data_nascimento" < CURRENT_TIMESTAMP);

```

```

ALTER TABLE "paciente"
ADD CONSTRAINT paciente_check_genero
CHECK ("genero" IN (1, 2, 3, 4, 5));
ALTER TABLE "paciente"
ADD CONSTRAINT paciente_check_estado_civil
CHECK ("estado_civil" IN (1, 2, 3, 4));

```

```

ALTER TABLE "agenda_semanal"
ADD CONSTRAINT agenda_semanal_check_dia_semana
CHECK ("dia_semana" IN (1, 2, 3, 4, 5, 6, 7));

```

```

ALTER TABLE "agenda_semanal"
ADD CONSTRAINT agenda_semanal_check_horario
CHECK ("horario_inicio" < "horario_fim");

```

```
ALTER TABLE "restricoes_horario"  
  ADD CONSTRAINT restricoes_horario_check_horario  
  CHECK ("horario_inicio" < "horario_fim");
```

```
ALTER TABLE "restricoes_horario"  
  ADD CONSTRAINT restricoes_horario_check_data  
  CHECK ("data" + "horario_inicio" > CURRENT_TIMESTAMP);
```

```
ALTER TABLE "restricoes_agenda_semanal"  
  ADD CONSTRAINT restricoes_agenda_semanal_check_horario  
  CHECK ("horario_inicio" < "horario_fim");
```

```
ALTER TABLE "consulta"  
  ADD CONSTRAINT consulta_check_situacao  
  CHECK ("situacao" IN (1, 2, 3, 4, 5));
```

```
ALTER TABLE "profissional"  
  ADD CONSTRAINT profissional_duracao_base_check  
  CHECK ("duracao_base" > 0);
```



## 9 CRONOGRAMA

Atividades	30/05 à 10/06	13/06 à 24/06	27/06 à 01/07	04/07 à 15/07	18/07 à 29/07	01/08 à 12/08	15/08 à 02/09	05/09 à 16/09	19/09 à 30/09	03/10 à 14/10	17/10 à 28/10	31/10 à 11/11	14/11 à 25/11	28/11 à 09/12
Confecção de anteprojeto.														
Modelagens, protótipos de interface.														
Validação do cliente e ajustes finais.														
Início dos projetos de <i>software</i> .														
Modelagem plano de teste e protótipos de interface														
Codificação dos projetos <i>frontend web</i> e <i>backend</i> .														
Codificação dos projetos <i>frontend web</i> e <i>backend</i> .														
Codificação dos projetos <i>frontend web</i> e <i>backend</i> .														
Codificação dos projetos <i>frontend web</i> e <i>backend</i> .														
Codificação dos projetos <i>frontend web</i> e <i>backend</i> .														
Codificação dos projetos <i>frontend web</i> e <i>backend</i> .														
Execução do plano de testes.														
Confecção do <i>Help</i> .														
Implantação do sistema computacional.														
Treinamento.														

Tarefas realizadas

Tarefas a realizar

## **10 CONCLUSÃO**

Citados os principais pontos para compreensão do projeto, conclui-se que seu desenvolvimento, seguindo o modelo descrito aqui, acarretará benefícios para o cliente, como, por exemplo, economia de tempo, praticidade no gerenciamento e segurança dos dados, ainda, para os desenvolvedores, trará experiência de desenvolvimento com um produto real, pavimentando o caminho para entrada no mercado de trabalho.

## 11 REFERÊNCIAS

- 1 PRESSMAN, Roger S.; MAXIM, Bruce R.. **Engenharia de software: uma abordagem profissional**. 8 ed. Porto Alegre: AMGH, 2016.
- 2 Salário. **Programador de Computador - Salário 2022 - Ponta Grossa, PR - Mercado de Trabalho**. Disponível em: <https://www.salario.com.br/profissao/programador-de-computador-cbo-317110/ponta-grossa-pr/>. Acesso em: 05 de jun. 2022.
- 3 FREIRE H.. **Calculando Estimativas: o Método de Pontos de Caso de Uso**. Disponível em: <https://www.cin.ufpe.br/~raa3/projetao/2Iteracao/Apresentacao/usecasepoints.pdf>. Acesso em: 05 de jun. 2022.
- 4 FaCom - UFU. **Estimativas baseadas em casos de uso**. Disponível em: <https://www.facom.ufu.br/~bacala/ES/07%20-%20Metricas%20UseCases.pdf>. Acesso em: 05 de jun. 2022.

## 12 RESPONSABILIDADES

---

Marcos Renan Krul

---

Renato Cristiano Ruppel

## APÊNDICE A - SCRIPT PARA CRIAÇÃO DE PAPÉIS DE USUÁRIO E PERMISSÕES.

```

INSERT INTO "papel_usuario" ("nome") VALUES('OWNER');
INSERT INTO "papel_usuario" ("nome") VALUES('PROFESSIONAL');
INSERT INTO "papel_usuario" ("nome") VALUES('EMPLOYEE');
INSERT INTO "papel_usuario" ("nome") VALUES('PROFESSIONAL_UNCONFIGURED');

INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'READ_APPOINTMENTS');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'CREATE_PROFESSIONAL');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'READ_PROFESSIONAL');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'UPDATE_PROFESSIONAL');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'CREATE_EMPLOYEE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1, 'READ_EMPLOYEE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'UPDATE_EMPLOYEE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1, 'CREATE_PATIENT');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1, 'READ_PATIENT');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1, 'UPDATE_PATIENT');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'UPDATE_PROFESSIONAL');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'DELETE_PROFESSIONAL');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1,
'DELETE_EMPLOYEE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1, 'DELETE_PATIENT');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(1, 'READ_LIABLE');

INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'UPDATE_PROFESSIONAL');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'CREATE_WEEKLY_SCHEDULE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'READ_WEEKLY_SCHEDULE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'UPDATE_WEEKLY_SCHEDULE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'CREATE_WEEKLY_SCHEDULE_LOCK');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'DELETE_WEEKLY_SCHEDULE_LOCK');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'READ_WEEKLY_SCHEDULE_LOCK');

```

```

INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'UPDATE_WEEKLY_SCHEDULE_LOCK');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'CREATE_SCHEDULE_LOCK');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'READ_SCHEDULE_LOCK');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'CREATE_COMMENTS');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'READ_COMMENTS');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2,
'DELETE_SCHEDULE_LOCK');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(2, 'READ_PATIENT');

```

```

INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3,
'READ_APPOINTMENTS');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3,
'UPDATE_APPOINTMENTS');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3,
'READ_PROFESSIONAL');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3, 'CREATE_PATIENT');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3, 'READ_PATIENT');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3, 'UPDATE_PATIENT');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3,
'READ_WEEKLY_SCHEDULE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3,
'READ_WEEKLY_SCHEDULE_LOCK');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3,
'READ_SCHEDULE_LOCK');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3, 'DELETE_PATIENT');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3, 'READ_LIABLE');
INSERT INTO "permissoes" ("id_papel_usuario", "nome") VALUES(3,
'CREATE_APPOINTMENT');

```