

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

Bacharelado em Engenharia de Software - 4º Período

DISCIPLINA: *Oficina de Integração 1 - ES63G-ES41*

PROFESSOR: *Gabriel Canhadas Genvigir*

Documento de Projeto de Software

Galenus

Amanda Beatriz Machado

Ana Carolina Moreira da Silva

Arthur Henrique Abrantes Hassimyan

Augusto Baroni Simionato

Patrik Melo Dias

Renato Sardinha Lopes

Cornélio Procópio

2023

LISTA DE ILUSTRAÇÕES

Figura 1: Processo de desenvolvimento de software - Scrum	12
Figura 2: Caso de uso geral do sistema Galenus.....	20
Figura 3: Modelo Conceitual DER do Sistema Galenus	21
Figura 4: Modelo lógico do Sistema Galenus.....	22
Figura 5: Diagrama de classes do sistema	26
Figura 6: Diagrama de atividade agendar consulta.....	27
Figura 7: Tela inicial do sistema Galenus.....	28
Figura 8: Tela agendar consulta.....	29
Figura 9: Design System do sistema Galenus.....	30
Figura 10: Organização de pacotes e classes do projeto - NetBeans	31
Figura 11: Diagrama de pacotes do sistema	32
Figura 12: Diagrama de componentes do sistema	32
Figura 13: Módulo gerenciar agenda	38
Figura 14: Módulo gerenciar funcionários	38
Figura 15: Módulo gerenciar pacientes.....	39
Figura 16: Módulo gerenciar prontuário.....	39
Figura 17: Módulo fazer login	40
Figura 18: Diagrama de atividades alterar consulta.....	41
Figura 19: Diagrama de atividades cancelar consulta.....	41
Figura 20: Diagrama de atividades visualizar consulta	41
Figura 21: Diagrama de atividades alterar funcionário.....	42
Figura 22: Diagrama de atividades visualizar funcionário	42
Figura 23: Diagrama de atividades cadastrar funcionários.....	42
Figura 24: Diagrama de atividades alterar pacientes	43
Figura 25: Diagrama de atividades cadastrar pacientes	43
Figura 26: Diagrama de atividades visualizar pacientes.....	43
Figura 27: Diagrama de atividades visualizar prontuário	44
Figura 28: Diagrama de atividades registrar no prontuário	44
Figura 29: Diagrama de atividades realizar login	44
Figura 30: Tela principal para os funcionários dos Recursos Humanos (RH)	45
Figura 31: Tela de cadastro de funcionários	45

Figura 32: Tela cadastrar médico.....	46
Figura 33: Tela alterar cadastro	46
Figura 34: Tela visualizar cadastro	47
Figura 35: Tela principal da recepção.....	47
Figura 36: Tela confirmar consulta.....	48
Figura 37: Tela agendar consulta.....	48
Figura 38: Tela cadastrar pacientes.....	49
Figura 39: Tela cancelar consulta	49
Figura 40: Tela área do médico	50
Figura 41: Tela do prontuário do paciente	50
Tabela 1: Comparação dos dois casos	6
Tabela 2: Time Scrum do projeto Galenus	12
Tabela 3: Cronograma previsto.....	13
Tabela 4: Requisitos funcionais	17
Tabela 5: Requisitos não-funcionais.....	19
Tabela 6: Dicionário de dados	23

Sumário

1	INTRODUÇÃO	5
1.1	Contexto	5
1.2	Justificativa.....	7
1.3	Proposta.....	8
1.4	Organização do Documento	8
2	DESCRIÇÃO GERAL DO SISTEMA	9
2.1	Objetivos (Gerais e Específicos)	9
2.2	Limites e Restrições	9
2.3	Descrição dos Usuários do Sistema	10
3	DESENVOLVIMENTO DO PROJETO	10
3.1	Tecnologias e Ferramentas.....	10
3.2	Metodologia de Desenvolvimento	11
3.3	Cronograma Previsto.....	13
4	REQUISITOS DO SISTEMA	17
4.1	Requisitos Funcionais	17
4.2	Requisitos Não-funcionais	19
4.3	Diagramas de Casos de Uso	20
5	ANÁLISE DO SISTEMA.....	21
5.1	Modelo Conceitual do Banco de Dados	21
5.2	Modelo Lógico do Banco de Dados	22
5.3	Dicionário de Dados	23
5.4	Diagrama de Classes	26
5.5	Diagrama de Atividades	27
6	IMPLEMENTAÇÃO	28
6.1	Protótipos de Telas	28
6.2	Descrição do Código	30
7	CONSIDERAÇÕES FINAIS	34
8	BIBLIOGRAFIA	36
	APÊNDICES	37

1 Introdução

1.1 Contexto

Em decorrência dos anos, juntamente com seus avanços e aprimoramentos tecnológicos, a maioria das empresas foram se adequando a essa nova realidade digital, otimizando sua forma de trabalho em diversos setores inclusive na gestão organizacional, trazendo qualidade, retorno financeiro, segurança, agilidade, apoio às decisões e assim contemplando todo o gerenciamento das informações no ambiente empregado, conseqüentemente, da mesma forma ocorreu com as clínicas médicas que também passaram por todo esse processo de adaptação, assim, no passado, o que antes era voltado a arquivos e pranchetas, atualmente vem sendo fortemente substituído por sistemas de software médicos.

Em um estudo realizado em 2011 pelo VIII Simpósio de Excelência em Gestão e Tecnologia, que tem como propósito de pesquisa a comparação entre dois hospitais, avaliou-se a discrepância presente entre os tipos de gestões hospitalares. Sendo o primeiro uma Santa Casa de Misericórdia, de médio porte, que utiliza da atividade manual na recepção, no atendimento médico, na atividade administrativa de preenchimento de fichas, e o outro um Hospital Particular de médio porte, que diferentemente da Santa Casa supracitada, utiliza um sistema integrado predominante digital que acumula todos os registros referentes às pessoas sob cuidados médicos e seu atendimento, possuindo desta forma a ausência de pessoas caminhando com documentos e informações pessoais de pacientes e colaboradores.

O estudo de caso e os dados coletados, mostraram que a falta de aprimoramentos tecnológicos ocasionam em um desperdício de tempo significativo por parte dos funcionários que além de utilizarem um sistema manual, ainda se deslocam de seus postos para exercerem outras funções que seriam substituídas por um sistema informatizado, fazendo com que o tempo de espera dos pacientes para atendimento médico aumente consideravelmente, além disso, é válido salientar ainda, que devido a essa gestão os médicos encontram uma certa dificuldade de acesso ao histórico de seus pacientes podendo até mesmo interferir em um diagnóstico mais preciso.

De uma visão mais quantitativa, o estudo abordado gerou dados que comparam o cenário administrativo nos dois hospitais.

Tabela 1: Comparação dos dois casos

Siglas	<i>Santa Casa</i>	<i>Hospital Particular</i>	<i>Evolução</i>
Funcionários	98	90	-8,2%
Corpo Clínico	25	20	-20,0%
Atendimentos	250	300	20,0%
Espera	até 4 horas	até 2,5 horas	-37,5%
Atendimento por médico	10	15	50,0%
Espera por atendimento	1000	750	-25,0%

Fonte: LEITE; LAURETTI, 2011

Considerando as estatísticas apresentadas, em um hospital onde possui um sistema de software integrado, a quantidade de funcionários caiu em 8,2%, assim como o Corpo Clínico em 20%, ato este causado pela melhoria na eficiência na empresa, eliminando redundância de tarefas e melhorando os processos. Em compensação, o número de atendimentos e espera tiveram melhoras significativas, os atendimentos subiram em 20% e a espera que na Santa Casa o paciente poderia aguardar até 4 horas, no Hospital Particular há uma diminuição de 37,5% ou seja, o atendimento ocorre em até 2,5 horas, o estudo demonstra também que os atendimentos por médico no Hospital Particular subiram em torno de 50%, outro progresso importante que mostram os benefícios de um sistema de software médico integrado nos ambientes hospitalares.

A conclusão da pesquisa foi de que a tecnologia no ambiente hospitalar traz um processo enxuto e rápido, obtendo ganhos no número de pessoas envolvidas com informações compartilhadas, número de etapas suscetíveis a falhas, qualidade da informação compartilhada, além da confiabilidade e agilidade no atendimento aos pacientes, culminando no fato de que todos estes pontos projetam melhoria do resultado financeiro, o que garante a sustentabilidade hospitalar, ampliação da segurança das informações confidenciais do paciente e também a prevenção dos danos causados pelo desgaste ou possíveis acidentes.

Por outro lado, já na metodologia tradicional como a que ocorre na Santa Casa de Misericórdia, observa-se processos desnecessários com excesso de papéis, fluxo de informações relevantes que são confiados a apenas uma pessoa, e a movimentação física excessiva de todos os envolvidos, sendo com isso, todos estes pontos sujeitos a erros.

Dessa forma, em vista dos fatos supracitados, pode-se concluir que a tecnologia tem desempenhado um papel cada vez mais importante no campo da saúde, no entanto, com o avanço contínuo da medicina e também da tecnologia, há uma procura constante de novos softwares médicos capazes de atender às atuais técnicas e tratamentos, e que supram todas às necessidades dos usuários envolvidos. Além do mais, outra razão para o desenvolvimento de novos sistemas, é a evolução das expectativas dos pacientes, uma vez que, estão mais conscientes da importância da tecnologia na

prestação de cuidados da saúde e esperam que seus médicos utilizem as tecnologias mais recentes para fornecer o melhor atendimento possível.

Em consequência disso, atualmente vem crescendo exponencialmente a demanda de softwares para o gerenciamento administrativo em um ambiente hospitalar, existindo inclusive diversos sistemas no ramo, porém, ainda há espaço para a elaboração de sistemas cada vez mais otimizados, capazes de solucionar todos os problemas dos métodos tradicionais, trazendo mais performance e interfaces intuitivas para que qualquer profissional que o manipule seja capaz de entender sem um treinamento pré-estabelecido.

Por essa razão, para distanciar-se da gestão tradicional voltada a atividades manuais que ainda hoje é muito comum, e adentrar no mundo tecnológico, o presente projeto tem como objetivo desenvolver um sistema de gerenciamento administrativo para uma clínica médica, atendendo todas suas necessidades e expectativas, e fornecer uma maior segurança e qualidade a todos os usuários envolvidos.

1.2 Justificativa

A demanda por sistemas de softwares para o gerenciamento de clínicas médicas vem crescendo notavelmente. Isso se explica no fato de que os ambientes hospitalares estão cada vez mais com o intuito de diminuir as limitações que o modelo de arquivos físicos possui e também adentrar nesse novo mundo digital, o qual é bem visto pela atual sociedade.

Deste modo, a clínica médica de multi-especialidades Galenus como nova no mercado, ainda não possui um sistema informatizado para o seu gerenciamento, e buscando agregar mais valor e segurança, a clínica procura um sistema onde consegue realizar toda sua gestão organizacional de uma forma inovadora para levar mais qualidade e agilidade aos seus pacientes.

Portanto, o sistema de software Galenus, será desenvolvido para maximizar o gerenciamento hospitalar, em um ambiente informatizado onde todas as informações cruciais da instituição se concentram em um só lugar, realizando toda a gestão e controle. Desta forma, a clínica que atualmente não possui um suporte digital para o seu gerenciamento, se tornaria mais eficiente e eficaz, tanto para seus pacientes quanto para seus médicos e colaboradores.

1.3 Proposta

O presente projeto tem como proposta desenvolver uma aplicação desktop para auxiliar a clínica médica Galenus com sua gestão administrativa, através de um cadastramento detalhado de pacientes e médicos, consultas a históricos hospitalares, registro nos prontuários, agendamento e remarcação de consultas médicas, dentre outras funcionalidades que irá auxiliar todo o processo de informações, diminuindo o esforço dos funcionários e aumentando a qualidade do serviço oferecido.

1.4 Organização do Documento

O vigente documento de projeto de software visa detalhar toda a execução e aplicação utilizada no desenvolvimento do Sistema de Gerenciamento Galenus, assim como, registrar todos os artefatos gerados, diante disso, este documento está dividido em oito capítulos, sendo eles:

No primeiro capítulo deste documento é abordado a introdução do projeto, com o contexto em que o sistema vai ser inserido, a justificativa e a proposta para desenvolvimento do sistema, e o presente tópico, a organização do documento.

O segundo capítulo, apresenta a descrição geral do sistema, com seu objetivo geral e objetivos específicos, também aborda as limitações e restrições do sistema, além de uma descrição dos usuários que utilizarão o software.

No terceiro capítulo, denominado desenvolvimento do projeto, é apresentado as tecnologias e ferramentas utilizadas no sistema, assim como, a metodologia de desenvolvimento aplicada, e por fim o cronograma previsto do projeto.

No quarto capítulo, retrata sobre os requisitos do sistema, com um detalhamento dos requisitos funcionais e os requisitos não funcionais, além dos diagramas de casos de uso.

No quinto capítulo, é realizada uma análise do sistema, no qual são mostrados todos os artefatos gerados durante a execução do projeto como o modelo de banco de dados, diagrama de classes, diagrama de atividades, entre outros.

No sexto capítulo, é apresentada a implementação, com os protótipos de tela gerados e também com uma descrição detalhada do código desenvolvido.

No sétimo capítulo, é realizada as considerações finais onde é exposto os resultados obtidos e sua aplicabilidade.

Por fim, no oitavo capítulo estão as referências bibliográficas onde são detalhadas todas as fontes utilizadas no processo de desenvolvimento desse documento.

2 Descrição Geral do Sistema

2.1 Objetivos (Gerais e Específicos)

O projeto tem como objetivo realizar o gerenciamento administrativo da clínica médica Galenus, realizando todo o gerenciamento de funcionários e pacientes, assim como, o controle de consultas médicas e o registro de prontuários.

O software deve apresentar as seguintes funcionalidades:

- Gerenciar agendamento de consultas;
- Gerenciar documentos de pacientes;
- Gerenciar funcionários da clínica;
- Gerenciar pacientes da clínica;

2.2 Limites e Restrições

2.2.1 Limites

- Primeiramente o software não será desenvolvido para ambientes mobile e web;
- O software será desenvolvido exclusivamente para o ambiente de uma clínica médica, necessitando estar instalado em pelo menos um computador na sala de recepção e pelo menos um computador em cada sala destinada ao médico, conectados localmente por uma rede ethernet;
- O médico poderá apenas disponibilizar seus horários de atendimento para marcação de consultas, adicionar informações ao prontuário e buscar por consultas agendadas.
- A pessoa responsável pela recepção poderá apenas marcar consultas a partir dos horários disponibilizados pelo médico e anotar as informações no cadastro do paciente.

2.2.2 Restrições

- O software será restrito ao ambiente desktop executado por uma aplicação Java, sendo assim, é necessário ter instalado o Java Runtime Environment (JRE).
- O software necessita de conexão com a rede local e acesso à internet.
- O software necessita de um servidor SQL para fazer as atualizações no banco de dados.
- O médico poderá marcar consultas apenas dentro do horário de funcionamento da clínica e o mesmo se aplica à pessoa responsável pela recepção.
- O software não emitirá receitas médicas, pertencendo esta responsabilidade ao médico.
- O software só permitirá o cancelamento de consultas médicas com no mínimo 2 horas de antecedência.

2.3 Descrição dos Usuários do Sistema

Os usuários do sistema serão recepcionistas, médicos e ocasionalmente os funcionários do Recursos Humanos, ou seja, pessoas que trabalham na clínica na qual será implementada a solução e que irão utilizar o sistema no dia a dia para o gerenciamento de consultas, funcionários e pacientes.

3 Desenvolvimento do Projeto

3.1 Tecnologias e ferramentas

- **Linguagem de programação:** Java 19.0.2 utilizando biblioteca gráfica do JavaFx.
- **Modelagem:** Astah UML versão 9.0.0/1778f1;
- **Banco de Dados:** MySQL Workbench versão 8.0.32 utilizando o servidor online db4free;
- **Ferramentas de Desenvolvimento:**
 - **IDE:** Apache NetBeans IDE 17;
 - **Runtime:** Java (TM) SE Runtime Environment 19.0.2+7-44.
- **Organização da divisão de tarefas:** Trello e WhatsApp;
- **Organização do documento e controle de versões:** Pasta Compartilhada do Projeto no Google Drive, Google Docs. e GitHub para versionamento do sistema.
- **Prototipação:** Figma.

3.2 Metodologia de desenvolvimento

O Scrum é uma metodologia ágil de desenvolvimento de projetos que ajuda as equipes a estruturar e gerenciar o trabalho por meio de um conjunto de valores, princípios e práticas que fornecem valor significativo rapidamente em todo o projeto.

Sendo um dos frameworks ágeis mais populares, o Scrum possui uma estrutura adaptável, iterativa, rápida, flexível e eficaz que embora seja amplamente utilizado no desenvolvimento de projetos de software, os princípios e as lições dessa estrutura podem ser aplicados a todos os tipos de trabalhos em equipe.

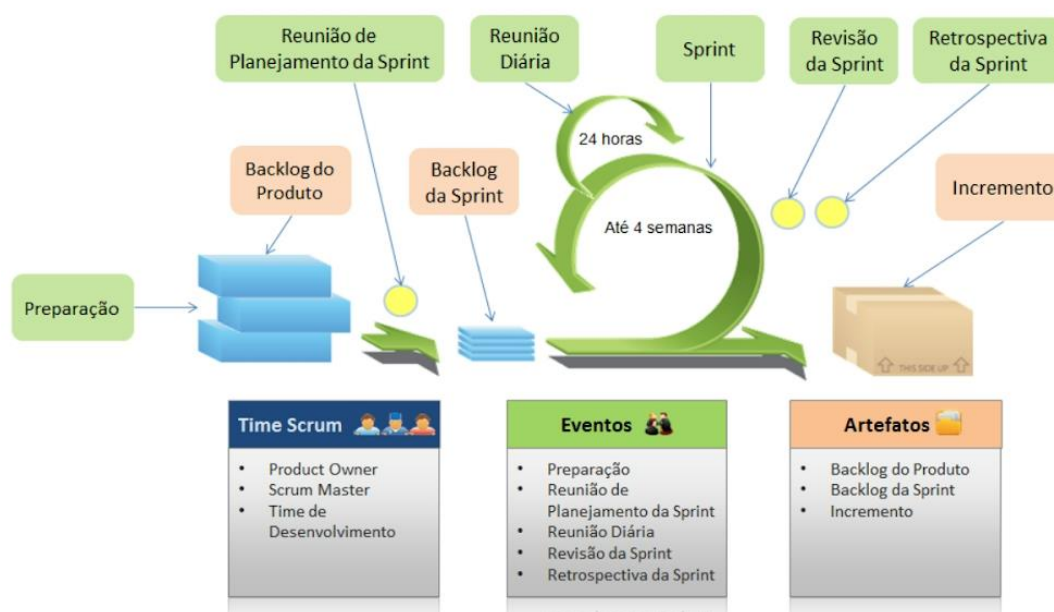
Um dos pontos fortes do Scrum está na utilização de times multifuncionais, auto-organizados e empoderados que dividem o trabalho em ciclos curtos e concentrados chamadas Sprints.

O ciclo Scrum começa com uma Reunião de Stakeholders, durante a qual a Visão do Projeto é criada. O Dono do Produto então desenvolve um Backlog Priorizado do Produto que contém uma lista priorizada de requisitos de negócios e projetos escritos na forma de Histórias de Usuário.

Cada Sprint começa com uma Reunião de Planejamento da Sprint durante a qual as Histórias de Usuário de alta prioridade são consideradas para inclusão na Sprint, ao final dessa reunião é gerado o Backlog da Sprint. Uma Sprint geralmente dura entre uma e quatro semanas e envolve o Time Scrum trabalhando para criar entregáveis ou incrementos de produtos potencialmente passíveis de entrega. Durante a Sprint, são realizadas reuniões diárias curtas e altamente focadas de 15 minutos, nas quais os membros da equipe discutem o progresso diário. O Dono do Produto pode avaliar os entregáveis concluídos durante a Sprint e aceitar os que atendem aos Critérios de Aceitação predefinidos.

Perto do final da Sprint, uma Reunião de Revisão da Sprint é realizada durante a qual o Dono do Produto e principais business stakeholders recebem uma demonstração dos entregáveis. O ciclo da Sprint termina com uma Reunião Retrospectiva da Sprint, onde a equipe discute maneiras de melhorar os processos e o desempenho à medida que avançam para a Sprint subsequente. Além disso, ao final da Sprint, o Time de Desenvolvimento e o Product Owner demonstram o incremento do produto para os clientes do projeto e pessoas relevantes, na reunião de Revisão da Sprint, sendo o incremento um artefato gerado das novas funcionalidades e melhorias oriundas dos itens do Backlog do Produto implementadas na Sprint em questão.

Figura 1: Processo de desenvolvimento de software - Scrum



Fonte: Tribunal Regional do Trabalho do Paraná

Da mesma forma que citado anteriormente sobre o ciclo de um projeto adotando a metodologia Scrum, acontecerá com o sistema Galenus, porém com as seguintes especificidades:

1. Serão realizadas 3 Sprints de 4 semanas cada.
2. As reuniões diárias serão realizadas no mínimo duas vezes por semana.

Tabela 2: Time Scrum do projeto Galenus

TIME SCRUM	
Nome	Função
Amanda Machado	Product Owner
Patrik Melo	Scrum Master
Ana Moreira	Time de Desenvolvimento
Augusto Simionato	Time de Desenvolvimento
Arthur Hassimyan	Time de Desenvolvimento
Renato Lopes	Time de Desenvolvimento

Fonte: Autoria própria

3.3 Cronograma previsto

Nesta parte do documento será apresentado o cronograma de todo o processo de desenvolvimento do projeto utilizando a metodologia Scrum, nas seguintes tabelas estão contidas as informações sobre cada fase do Scrum, os eventos, os artefatos gerados, o período de tempo, e o responsável por cada atividade.

Tabela 3: Cronograma previsto

CRONOGRAMA PREVISTO				
FASE DE INICIALIZAÇÃO E PLANEJAMENTO DA SPRINT				
Evento	Atividade	Data	Responsável	Status
Preparação	Backlog do Produto	02/03/2023	Amanda	Concluído
Reunião de Planejamento	Backlog da Sprint	02/03/2023	Todos	Concluído
PRIMEIRA SPRINT – 1º SEMANA				
Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Contexto	08/03/2023→09/03/2023	Amanda	Concluído
Sprint/Reunião Diária	Justificativa	08/03/2023→09/03/2023	Augusto/Arthur	Concluído
Sprint/Reunião Diária	Proposta	08/03/2023→09/03/2023	Ana/Patrik	Concluído
Sprint/Reunião Diária	Organização do Documento	08/03/2023→09/03/2023	Renato	Concluído
PRIMEIRA SPRINT – 2º SEMANA				
Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Objetivos Gerais e específicos	15/03/2023→16/03/2023	Amanda/Renato	Concluído
Sprint/Reunião Diária	Limites e Restrições	15/03/2023→16/03/2023	Augusto/Arthur	Concluído
Sprint/Reunião Diária	Descrições dos Usuários do Sistema	15/03/2023→16/03/2023	Ana/Patrik	Concluído
PRIMEIRA SPRINT – 3º SEMANA				
Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Tecnologias e Ferramentas	22/03/2023→23/03/2023	Amanda/Renato	Concluído

Sprint/Reunião Diária	Metodologia de Desenvolvimento	22/03/2023→23/03/2023	Ana/Patrik	Concluído
Sprint/Reunião Diária	Cronograma Previsto	22/03/2023→23/03/2023	Augusto/Arthur	Concluído

PRIMEIRA SPRINT – 4º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Requisitos Funcionais	29/03/2023→30/03/2023	Amanda/Renato	Concluído
Sprint/Reunião Diária	Requisitos Não-Funcionais	29/03/2023→30/03/2023	Augusto/Arthur	Concluído
Sprint/Reunião Diária	Diagramas de Caso de Uso	29/03/2023→30/03/2023	Ana/Patrik	Concluído

FASE DE ENTREGA, REVISÃO E RETROSPECTIVA DA SPRINT

Evento	Atividade	Data	Responsável	Status
Revisão da Sprint	Backlog do Produto/ Incremento	05/04/2023→06/04/2023	Todos	Concluído
Retrospectiva da Sprint	Backlog do Produto	05/04/2023→06/04/2023	Todos	Concluído
Envio dos Entregáveis	Entrega do Documento	05/04/2023→06/04/2023	Todos	Concluído

PLANEJAMENTO DA SEGUNDA SPRINT

Evento	Atividade	Data	Responsável	Status
Reunião de Planejamento	Backlog da Sprint	12/04/2023	Todos	Concluído

SEGUNDA SPRINT – 5º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Modelo Conceitual do Banco de Dados	13/04/2023	Amanda/Renato	Concluído
Sprint/Reunião Diária	Modelo Lógico do Banco de Dados	13/04/2023	Augusto/Arthur	Concluído
Sprint/Reunião Diária	Dicionário de Dados	13/04/2023	Ana/Patrik	Concluído

SEGUNDA SPRINT – 6º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Continuação do Modelo Conceitual do BD	19/04/2023→20/04/2023	Amanda/Renato	Concluído

Sprint/Reunião Diária	Continuação do Modelo Lógico do BD	19/04/2023→20/04/2023	Augusto/Arthur	Concluído
Sprint/Reunião Diária	Continuação do Dicionário de Dados	19/04/2023→20/04/2023	Ana/Patrik	Concluído

SEGUNDA SPRINT – 7º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Diagrama de Atividades	26/04/2023→27/04/2023	Amanda/Renato	Concluído
Sprint/Reunião Diária	Diagrama de Classes	26/04/2023→27/04/2023	Augusto/Arthur	Concluído
Sprint/Reunião Diária	Diagrama de Atividades	26/04/2023→27/04/2023	Ana/Patrik	Concluído

SEGUNDA SPRINT – 8º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Continuação do Diagrama de Atividades	03/05/2023→04/05/2023	Amanda/Renato	Concluído
Sprint/Reunião Diária	Continuação do Diagrama de Classes	03/05/2023→04/05/2023	Augusto/Arthur	Concluído
Sprint/Reunião Diária	Continuação do Diagrama de Atividades	03/05/2023→04/05/2023	Ana/Patrik	Concluído

FASE DE ENTREGA, REVISÃO E RETROSPECTIVA DA SPRINT

Evento	Atividade	Data	Responsável	Status
Revisão da Sprint	Backlog do Produto/ Incremento	10/05/2023→11/05/2023	Todos	Concluído
Retrospectiva da Sprint	Backlog do Produto	10/05/2023→11/05/2023	Todos	Concluído
Envio dos Entregáveis	Entrega do Documento	10/05/2023→11/05/2023	Todos	Concluído

PLANEJAMENTO DA TERCEIRA SPRINT

Evento	Atividade	Data	Responsável	Status
Reunião de Planejamento	Backlog da Sprint	17/05/2023→18/05/2023	Todos	Concluído

TERCEIRA SPRINT - 9º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Protótipos de Telas	24/05/2023→25/05/2023	Augusto	Concluído

Sprint/Reunião Diária	Criação do Banco de Dados	24/05/2023→25/05/2023	Patrik	Concluído
Sprint/Reunião Diária	Implementação do Sistema	24/05/2023→25/05/2023	Arthur	Concluído

TERCEIRA SPRINT – 10º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Protótipos de Telas	31/05/2023→01/06/2023	Augusto	Concluído
Sprint/Reunião Diária	Continuação da Implementação do BD	31/05/2023→01/06/2023	Patrik	Concluído
Sprint/Reunião Diária	Continuação da Implem. do Sistema	31/05/2023→01/06/2023	Arthur	Concluído

TERCEIRA SPRINT – 11º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Continuação da Implem. do Sistema	07/06/2023→14/06/2023	Arthur	Concluído
Sprint/Reunião Diária	Descrição do Código	07/06/2023→14/06/2023	Renato/Ana	Concluído
Sprint/Reunião Diária	Testes de Software	07/06/2023→14/06/2023	Todos	Concluído
Sprint/Reunião Diária	Organização do Documento	07/06/2023→14/06/2023	Amanda	Concluído

TERCEIRA SPRINT – 12º SEMANA

Evento	Atividade	Data	Responsável	Status
Sprint/Reunião Diária	Testes e Validação do Sistema	15/06/2023	Todos	Concluído
Sprint/Reunião Diária	Organização do Documento	15/06/2023	Amanda	Concluído

FASE DE ENTREGA FINAL DO PROJETO

Evento	Atividade	Data	Responsável	Status
Revisão da Sprint	Revisão do Projeto	21/06/2023→22/06/2023	Todos	Concluído
Entrega	Entrega do Documento	21/06/2023→22/06/2023	Todos	Concluído
Entrega	Entrega do Sistema	21/06/2023→22/06/2023	Todos	Concluído

Fonte: Autoria própria

4 Requisitos do Sistema

4.1 Requisitos Funcionais

Os requisitos funcionais especificam as ações que o sistema deve ser capaz de executar, ou seja, as funções do sistema. Diante disso, na tabela a seguir listará todos os requisitos funcionais do sistema, com seus respectivos códigos de identificação, nível de prioridade e requisitos relacionados.

Classificação dos requisitos de acordo com sua prioridade:

- Essencial: Deve ser implementado para que o sistema funcione.
- Importante: Sem este requisito o sistema pode funcionar, mas não da maneira esperada.
- Desejável: Este tipo de requisito não compromete o funcionamento do sistema.

Tabela 4: Requisitos funcionais

REQUISITOS FUNCIONAIS			
ID	Funcionalidade	Prioridade	Relacionado
RF01	O sistema deve permitir ao usuário realizar o login.	Essencial	
RF02	O sistema deve permitir que o profissional de Recursos Humanos gerencie as informações dos funcionários, através do cadastramento, modificação e visualização dos dados.	Essencial	RF01
RF03	O sistema deve permitir ao médico/recepcionista o gerenciamento das informações do paciente, através do cadastramento, modificação e visualização dos dados.	Essencial	RF01
RF04	O sistema deve permitir restritamente ao médico o registro das informações da consulta no prontuário.	Importante	RF01, RF03
RF05	O sistema deve permitir restritamente ao médico a visualização do histórico e dados do paciente no prontuário médico.	Importante	RF01, RF03, RF04
RF06	O sistema deve permitir o agendamento e reagendamento de novas consultas médicas.	Essencial	RF01, RF03, RF07

RF07	O sistema deve permitir o acesso a agenda médica para visualização e também para o agendamento de novas consultas.	Essencial	RF01, RF03, RF06
------	--	-----------	------------------

Fonte: Autoria própria

Logo abaixo, será explicado detalhadamente todos os requisitos funcionais listados, descrevendo suas características e funcionalidades.

- RF01: Nessa funcionalidade, o usuário que poderá ser o médico, funcionário do RH, ou o recepcionista, para acessar o sistema deverá inserir e-mail e senha, e se ocorrer o login com sucesso, o usuário terá acesso ao sistema, do contrário, poderá tentar inserir seus dados novamente até conseguir o acesso.
- RF02: O profissional de Recursos Humanos é o único que consegue gerenciar todas as informações dos funcionários, deste modo, é ele que vai registrar um novo funcionário da clínica Galenus, modificar seus dados e ter acesso a visualização, não sendo possível, no entanto, realizar a exclusão dos dados, em caso de inatividade, o registro deve ser arquivado.
- RF03: As funcionalidades referentes aos dados de registro dos pacientes, serão disponíveis aos médicos e recepcionistas, sendo possível realizar o cadastramento de novos pacientes, modificação e visualização de seus dados, não sendo possível, no entanto, realizar a exclusão dos dados, em caso de inatividade, o registro deve ser arquivado.
- RF04: O prontuário médico é uma funcionalidade crucial para registrar as informações da consulta dos pacientes, a implementação dessa funcionalidade será para o registro do diagnóstico, identificação do paciente, registro de exames, prescrições médicas e dentre outras anotações que são necessárias durante a consulta. O prontuário médico é uma funcionalidade extremamente restrita ao médico, e a violação dessa condição, compromete diretamente na Lei Geral de Proteção a Dados nº 13.709, de 14 de Agosto de 2018.
- RF05: A funcionalidade de acesso ao histórico dos pacientes, o que nada mais é do que os registros de prontuários realizados em cada consulta, são de visualização exclusiva do médico.
- RF06: O sistema terá como função realizar o agendamento de novas consultas, mediante a inserção dos dados do paciente cruciais para a consulta e a escolha do horário disponível do médico responsável pelo atendimento. O sistema também permitirá em caso de reagendamento ou necessidade de troca de horários, marcar uma nova data e horário para a consulta.
- RF07: Será implementado no sistema uma agenda médica para facilitar a visualização da dos horários de cada médico para o agendamento e reagendamento de consultas.

4.2 Requisitos Não-funcionais

Os requisitos não-funcionais especificam as restrições que serão aplicadas sobre os serviços ou funções providas pelo sistema, eles expressam as condições que o software deve atender ou qualidades específicas que o software deve ter. Os requisitos não-funcionais estão relacionados às propriedades emergentes do sistema, como confiabilidade, desempenho, portabilidade, segurança e entre outras.

Na tabela a seguir será listado todos os requisitos não-funcionais do sistema Galenus, com seus respectivos códigos de identificação e sua categoria.

Tabela 5: Requisitos não-funcionais

REQUISITOS NÃO-FUNCIONAIS		
ID	Requisito	Categoria
RNF01	O sistema deve executar em qualquer sistema operacional.	Portabilidade
RNF02	A persistência das informações deve ser implementada, em um Sistema Gerenciador de Bancos de Dados Relacionais (MySQL).	Implementação
RNF03	O sistema deve ser robusto e tolerante a falhas, de modo que se houver algum erro os dados registrados devem ser recuperados pelo sistema.	Confiabilidade
RNF04	O sistema deve ser implementado na linguagem Java.	Implementação
RNF05	O sistema deve controlar o acesso às funcionalidades: Para controlar os dados dos funcionários as operações devem ser restritas ao funcionário do RH. Funcionalidades relacionadas a prontuários e informações do paciente devem ser restritas ao médico e operações relacionadas a agenda médica aos recepcionistas e médicos.	Segurança
RNF06	O sistema deve realizar o agendamento/reagendamento de consultas restritamente no horário de atendimento.	Segurança
RNF07	As consultas ao sistema devem ser respondidas em menos de três segundos.	Eficiência
RNF08	O sistema deve assegurar a privacidade das informações dos pacientes, funcionários e todos os usuários envolvidos de acordo com a Lei Geral de Proteção a Dados nº 13.709, de 14 de Agosto de 2018.	Ético
RNF09	O sistema precisa estar conectado à Internet para funcionar.	Disponibilidade
RNF10	O sistema só permitirá o cancelamento de consultas, com no mínimo 2 horas de antecedência do horário marcado.	Segurança

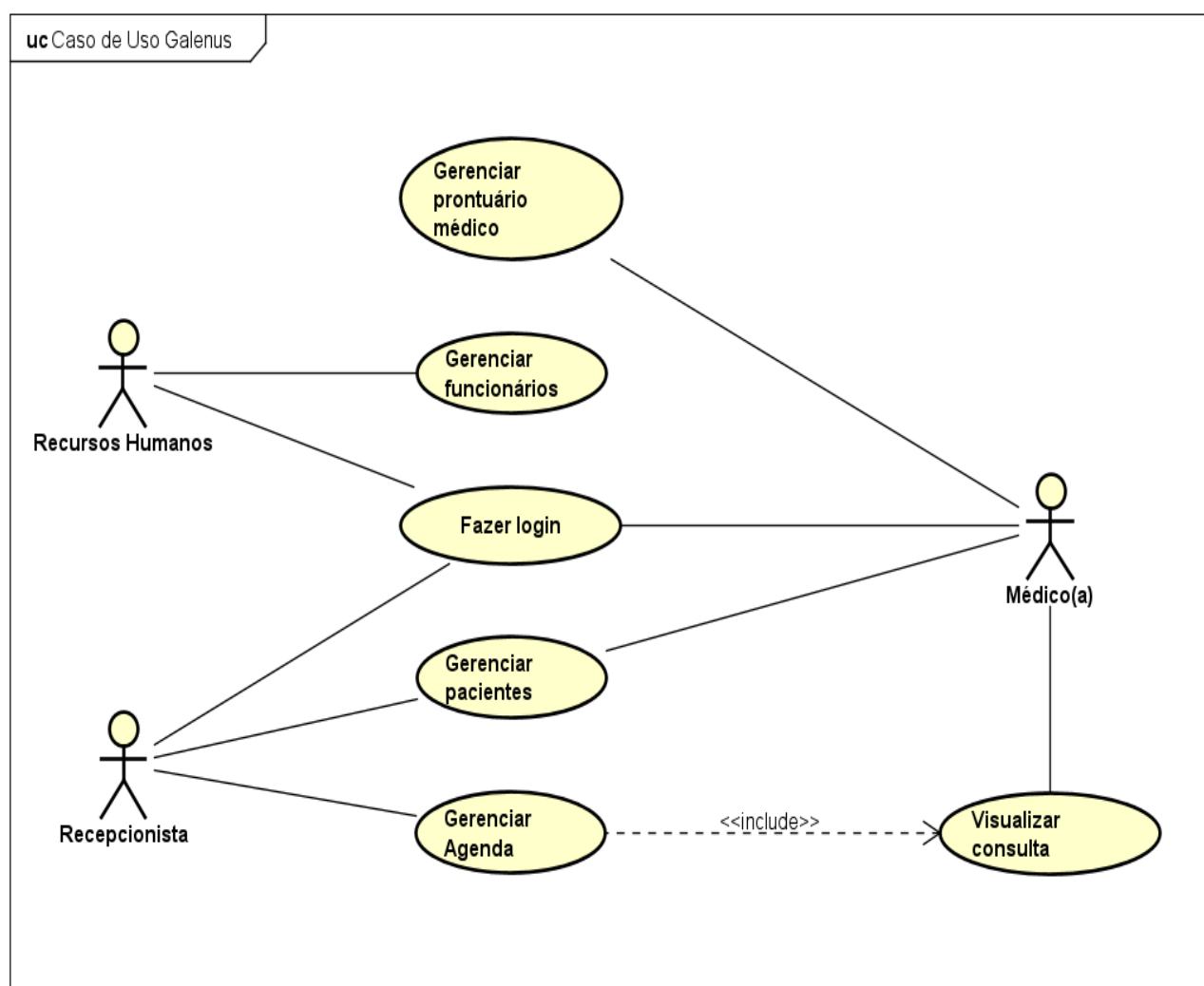
Fonte: Autoria própria

4.3 Diagramas de Casos de Uso

Na linguagem de modelagem unificada (UML), o diagrama de caso de uso apresenta uma visão externa geral das funcionalidades que o sistema deverá oferecer aos usuários, tentando identificar os tipos de usuários que irão interagir com o sistema, quais papéis irão assumir e quais funções um usuário específico poderá requisitar.

Logo abaixo será apresentado o diagrama com uma visão geral do sistema Galenus, com seus usuários e as funcionalidades permitidas pelo sistema, cada caso de uso contido no diagrama contém um módulo com suas funções específicas contidas no apêndice A.

Figura 2: Caso de uso geral do sistema Galenus



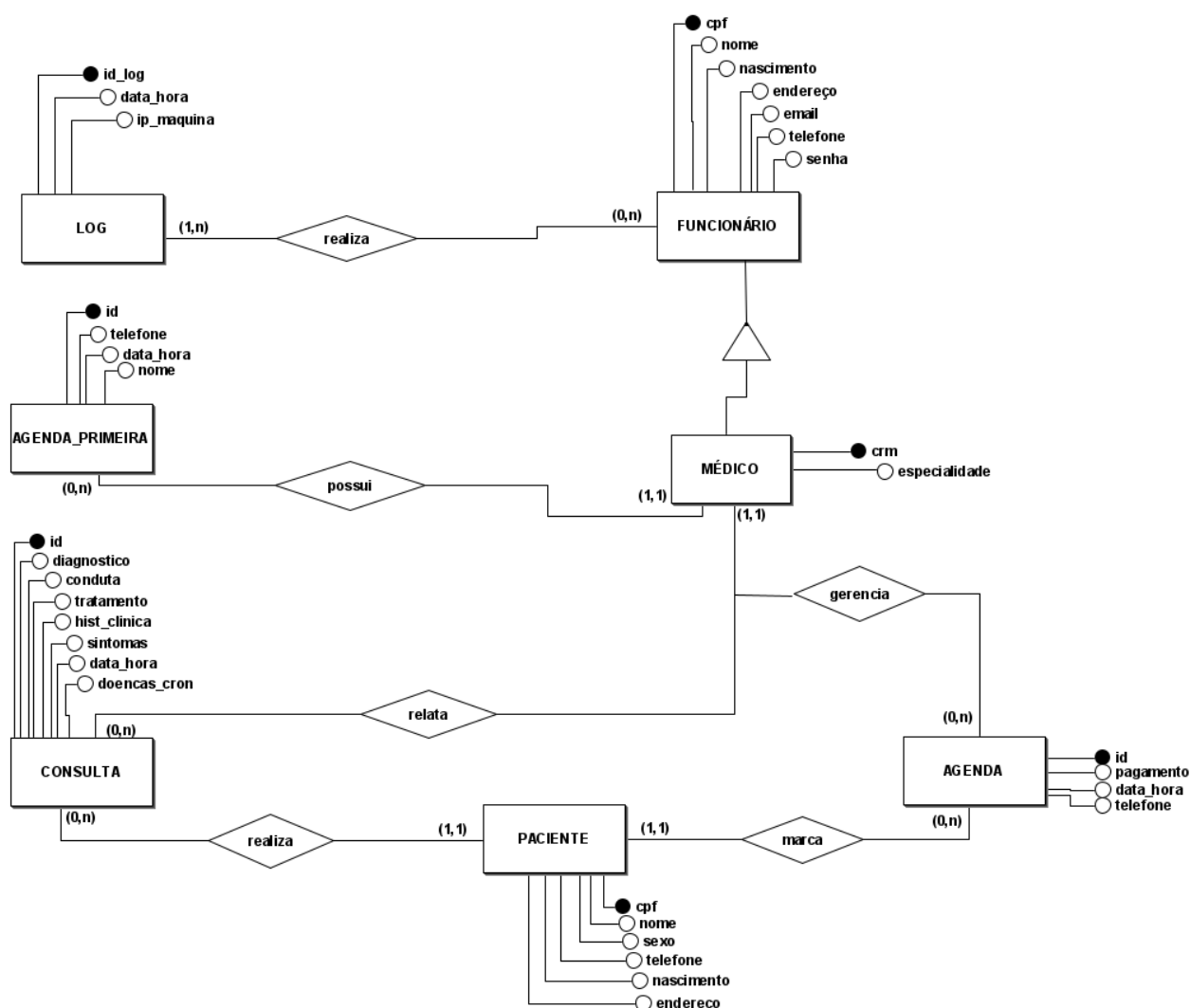
Fonte: Autoria própria

5 Análise do Sistema

5.1 Modelo Conceitual do Banco de Dados

O modelo DER (Diagrama de Entidade-Relacionamento) é o mais próximo da realidade dos usuários, sendo considerado um modelo conceitual construído com alto nível de abstração a partir dos requisitos do sistema extraídos na fase de levantamento de requisitos. Seu objetivo é especificar as entidades, seus atributos e os relacionamentos.

Figura 3: Modelo Conceitual DER do Sistema Galenus

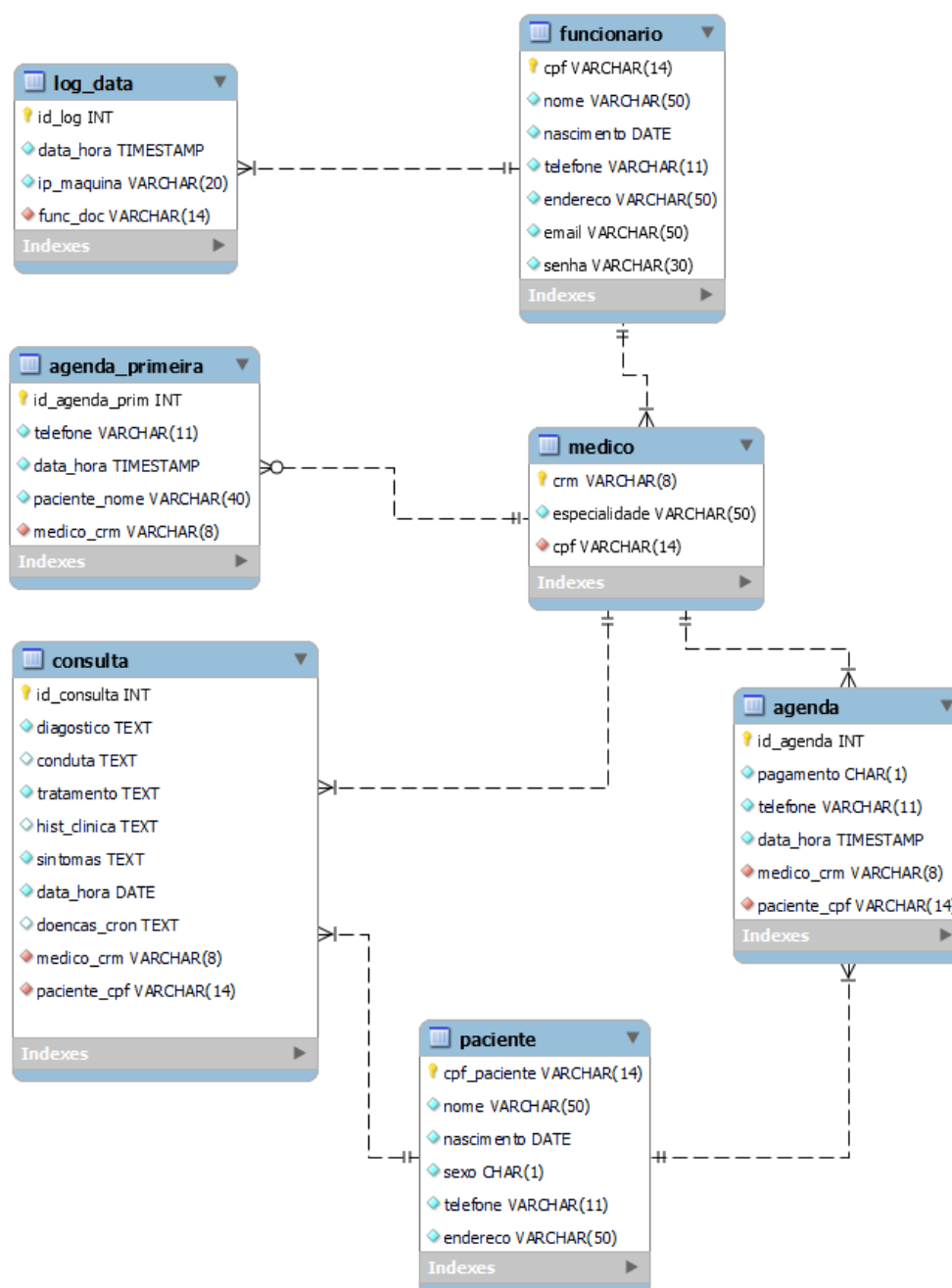


Fonte: Autoria própria

5.2 Modelo Lógico do Banco de Dados

O modelo de dados lógico é uma representação estruturada e formal dos dados que serão armazenados em um banco de dados, descrevendo mais detalhadamente as entidades, atributos e os relacionamentos. O modelo de dados lógico leva os elementos do modelo conceitual um passo adiante, adicionando mais informações a eles. Neste modelo, os conceitos abstratos são transformados em objetos concretos, como tabelas, campos e relacionamentos.

Figura 4: Modelo lógico do Sistema Galenus



Fonte: Autoria própria

5.3 Dicionário de Dados

Um dicionário de dados armazena e comunica informações sobre um banco de dados. É considerado uma lista de dados com os principais termos e métricas do banco de dados que está vinculado. Logo abaixo está o dicionário de cada tabela do sistema Galenus:

Tabela 6: Dicionário de dados

DICIONÁRIO DE DADOS				
Tabela: paciente				
Descrição: A tabela paciente contém todas as informações necessárias para registrar o paciente no banco de dados.				
Atributo	Descrição	Tipo	Nulo	Chave
cpf_paciente	Identificador do paciente (CPF)	VARCHAR(14)	Não	Primária(PK)
nome	Nome completo do paciente	VARCHAR(50)	Não	-
nascimento	Data de nascimento do paciente	DATE	Não	-
sexo	Sexo do paciente (F-Feminino/M-Masculino)	VARCHAR(1)	Não	-
telefone	Telefone/Celular de contato do paciente	VARCHAR(11)	Não	-
endereço	Endereço residencial do paciente	VARCHAR(50)	Não	-
Tabela: medico				
Descrição: A tabela medico, herda todas as informações da tabela funcionário e tem atributos específicos e essenciais do médico				
Atributo	Descrição	Tipo	Nulo	Chave
crm	Documento de identificação do médico.	VARCHAR(8)	Não	Primária(PK)
especialidade	Especialidade do médico.	VARCHAR(50)	Não	-
cpf	Identificador da tabela funcionário, que neste caso seria o documento CPF do médico.	VARCHAR(14)	Não	Estrangeira(FK)
Tabela: agenda				
Descrição: A tabela agenda contém todas as informações para realizar o agendamento de uma consulta.				
Atributo	Descrição	Tipo	Nulo	Chave
id_agenda	Identificador da tabela agenda	INT	Não	Primária(PK)
pagamento	Forma de pagamento do paciente	CHAR(1)	Não	-
telefone	Telefone/Celular de contato do paciente	VARCHAR(11)	Não	-
data_hora	Data e horário marcado da consulta	TIMESTAMP	Não	-

medico_crm	Identificador da tabela medico, que relaciona o médico selecionado para a consulta	VARCHAR(8)	Não	Estrangeira(FK)
paciente_cpf	Identificador da tabela paciente, que relaciona o paciente selecionado para a consulta	VARCHAR(14)	Não	Estrangeira(FK)

Tabela: funcionario

Descrição: A tabela funcionario contém todas as informações necessárias para realizar o cadastro de um funcionário. Os funcionários são todos os colaboradores da clinica, dentre eles, médicos, RH e recepcionistas

Atributo	Descrição	Tipo	Nulo	Chave
cpf	Identificador do funcionário (CPF)	VARCHAR(14)	Não	Primária(PK)
nome	Nome completo do funcionário	VARCHAR(50)	Não	-
nascimento	Data de nascimento do funcionário	DATE	Não	-
telefone	Telefone/Celular de contato do funcionário	VARCHAR(11)	Não	-
endereço	Endereço residencial do funcionário	VARCHAR(50)	Não	-
email	E-mail de acesso do funcionário	VARCHAR(50)	Não	-
senha	Senha de acesso do funcionário	VARCHAR(30)	Não	-

Tabela: log_data

Descrição: A tabela log_data contém informações sobre os logins realizados pelos funcionários.

Atributo	Descrição	Tipo	Nulo	Chave
id_log	Identificação da tabela log_data	INT	Não	Primária(PK)
data_hora	Data e horário da realização do login	TIMESTAMP	Não	-
ip_maquina	Identificação da máquina do usuário	VARCHAR(20)	Não	-
func_doc	Identificação da tabela funcionario, que neste caso, identifica o funcionário que realizou o login	VARCHAR(14)	Não	Estrangeira(FK)

Tabela: agenda_primeira

Descrição: A tabela agenda_primeira contém informações temporárias dos pacientes e necessárias para rápido agendamento de consultas (Ex: Telefonema), e que serão confirmadas posteriormente na tabela agenda.

Atributo	Descrição	Tipo	Nulo	Chave
id_agenda_prim	Identificação da tabela agenda_primeira	INT	Não	Primária(PK)
telefone	Telefone/Celular do paciente	VARCHAR(11)	Não	-
data_hora	Data e horário marcado da consulta	TIMESTAMP	Não	-
paciente_nome	Nome do paciente que irá realizar a consulta	VARCHAR(40)	Não	-

medico_crm	Identificador da tabela medico, que relaciona ao médico selecionado para a consulta	VARCHAR(8)	Não	Estrangeira(FK)
-------------------	---	------------	-----	-----------------

Tabela: consulta

Descrição: A tabela consulta contém todas as informações resultante do atendimento do médico com o paciente

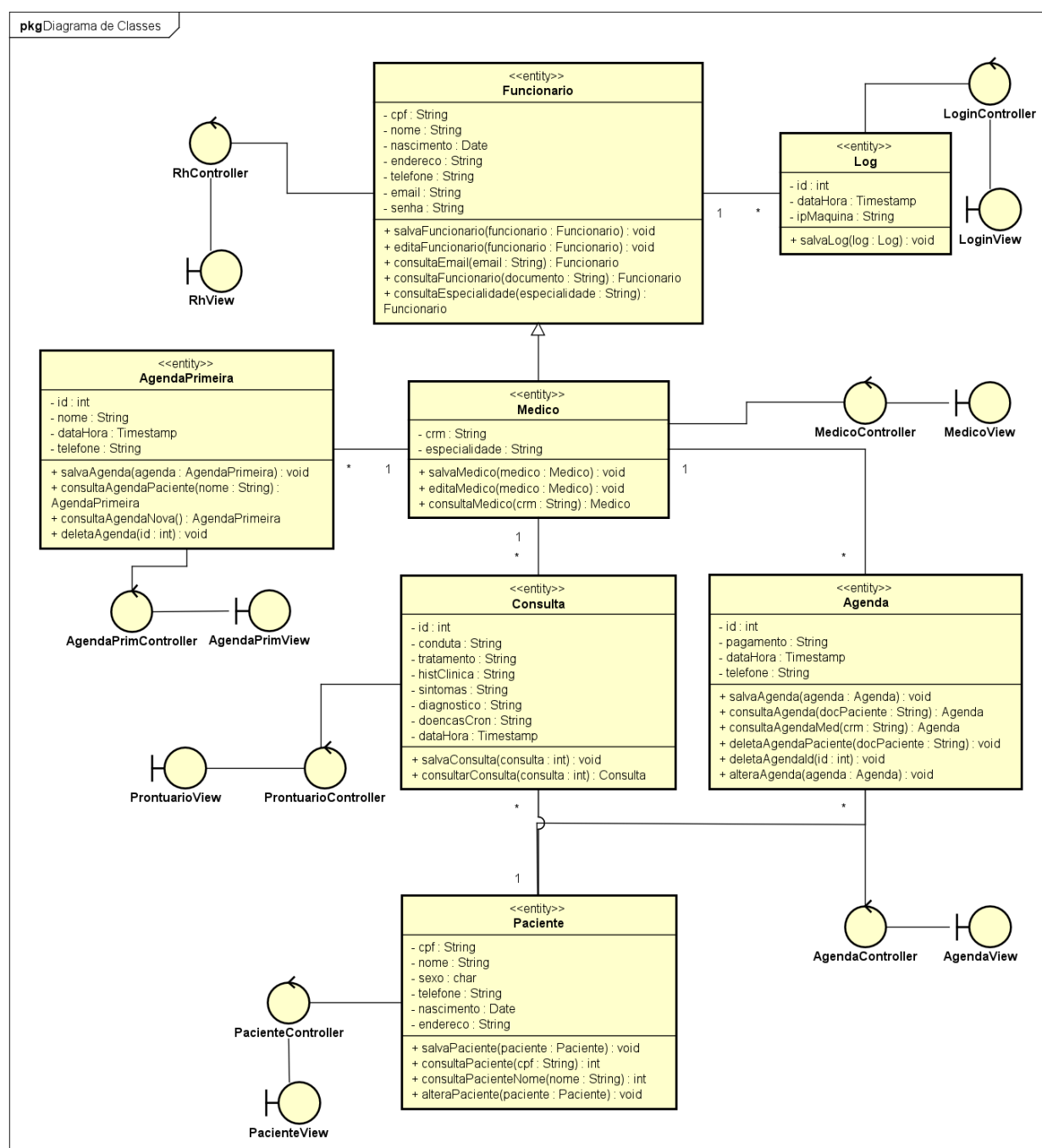
Atributo	Descrição	Tipo	Nulo	Chave
id_consulta	Identificação da tabela consulta	INT	Não	Primária
conduta	Plano de tratamento estabelecido	TEXT	Sim	-
tratamento	Tratamento estabelecido para o paciente	TEXT	Não	-
hist_clinica	Histórico do paciente	TEXT	Sim	
sintomas	Sintomas do paciente	TEXT	Não	-
data_hora	Data e horário da consulta	DATE	Não	-
diagnostico	Diagnóstico do paciente	TEXT	Não	-
doencas_cron	Doenças crônicas diagnosticadas	TEXT	Sim	-
medico_crm	Identificador da tabela medico, que relaciona o médico responsável pela realização da consulta.	VARCHAR(8)	Não	Estrangeira(FK)
paciente_cpf	Identificador da tabela paciente, que relaciona o paciente a ser consultado.	VARCHAR(14)	Não	Estrangeira(FK)

Fonte: Autoria própria

5.4 Diagrama de Classes

O diagrama de classes é uma representação essencial da estrutura das classes em um sistema, fazendo parte da Linguagem de Modelagem Unificada (UML). Ele define os atributos e métodos de cada classe, além de estabelecer como as classes se relacionam e trocam informações entre si. No contexto do sistema Galenus, foi desenvolvida essa diagramação com o intuito de proporcionar uma visão mais detalhada da composição do sistema, conforme demonstrado na figura abaixo:

Figura 5: Diagrama de classes do sistema



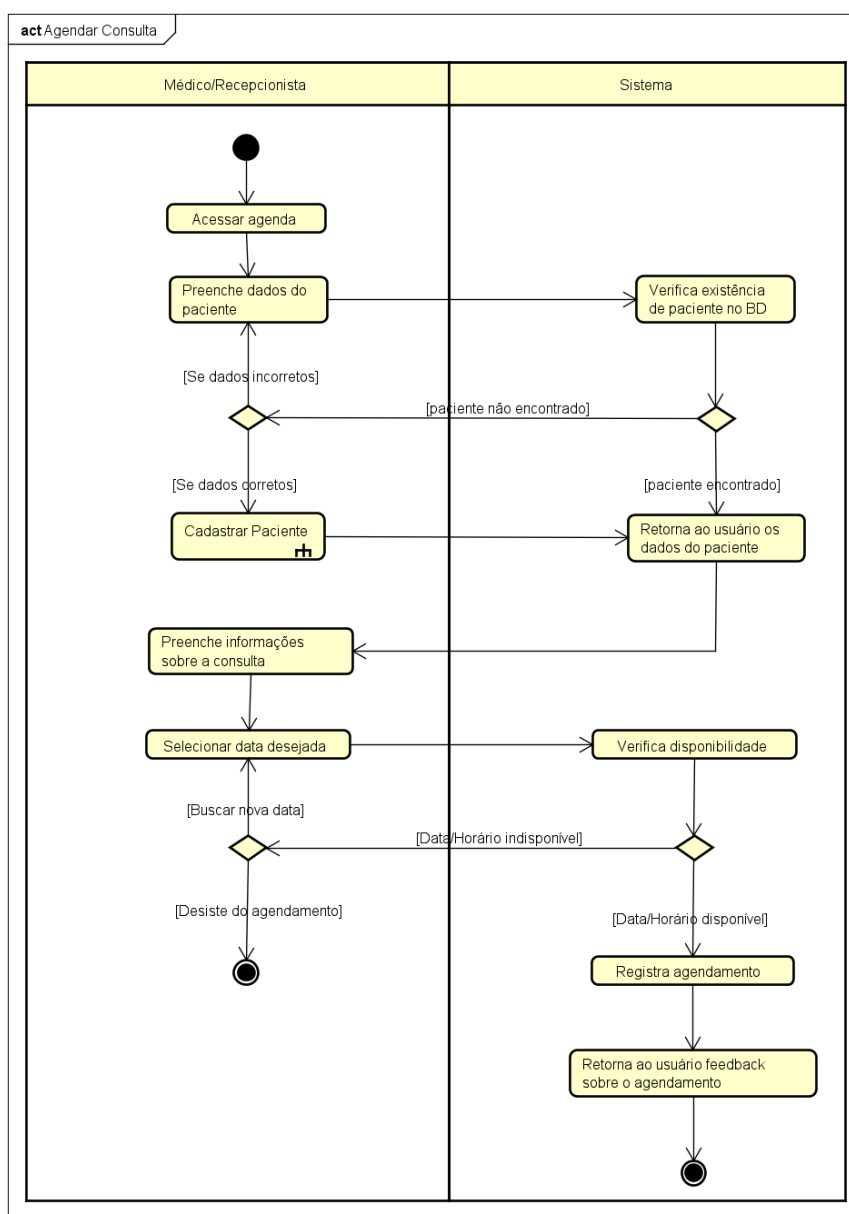
Fonte: Autoria própria

5.5 Diagrama de Atividades

Um diagrama de atividades mostra o fluxo de atividades em um sistema ou processo, utilizado para descrever aspectos dinâmicos e comportamentais do sistema. Eles são representações gráficas de fluxos de trabalho e ações passo a passo, usados para modelar processos computacionais e organizacionais, bem como fluxos de dados que se intersectam com eles.

O diagrama apresentado abaixo mostra o fluxo necessário para realizar a atividade de agendar uma consulta médica pelo usuário requisitado, as demais atividades proporcionadas pelo sistema Galenus estarão contidas no Apêndice B.

Figura 6: Diagrama de atividade agendar consulta



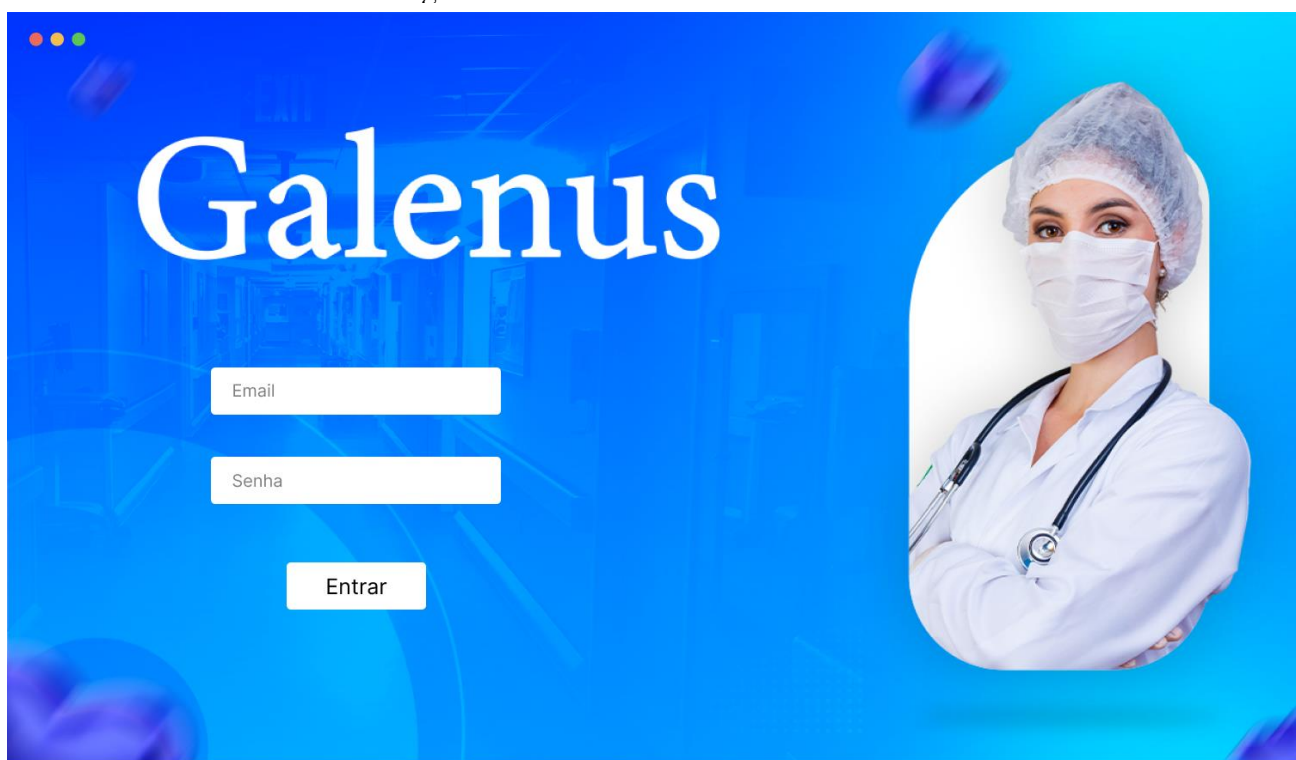
Fonte: Autoria própria

6 Implementação

6.1 Protótipos de Telas

A prototipação de telas é um processo essencial para criar representações visuais interativas de um produto ou sistema. Seu objetivo principal é visualizar, testar as funcionalidades, validar as ideias, o fluxo de navegação e a usabilidade antes do desenvolvimento final. Desse modo, para o sistema em questão utilizamos o Figma, um editor gráfico de vetor e prototipagem de projetos de design para as interfaces do sistema Galenus. Logo abaixo, será apresentada a tela inicial do sistema Galenus e a tela de agendamento, estando as demais contidas no Apêndice C.

Figura 7: Tela inicial do sistema Galenus



Fonte: Autoria própria

A primeira tela do sistema como demonstrada na figura acima, tem por objetivo disponibilizar o acesso ao sistema para os funcionários da clínica quando o sistema é iniciado. Nela há, além do título, dois campos de entrada, sendo um para senha e o outro para o e-mail, e um botão "Entrar" para efetuar o login. Ao clicar no botão, o usuário será direcionado para a tela inicial relacionada ao @ utilizado no campo de "Email".

Outra tela que podemos exemplificar é a de agendamento de consultas. Nela, o funcionário responsável pelo agendamento insere todas as informações nos campos em branco, como nome, telefone, médico, especialidade, data e hora. Após o preenchimento, basta clicar em "Agendar" para realizar o agendamento com sucesso, caso a data/hora e o médico(a) estejam disponíveis, do contrário, é possível tentar agendar novamente. Além disso, há dois botões extras: "Voltar" para retornar à tela anterior e "Limpar" para apagar os campos e fazer um novo preenchimento em caso de erros.

Figura 8: Tela agendar consulta

A imagem mostra a interface de usuário para agendar uma consulta. O título "Agendar Consulta" está centralizado no topo. Abaixo dele, há seis campos de entrada: "Nome" e "Telefone" (campos de texto simples), "Médico" e "Especialidade" (menus suspensos), e "Data" e "Hora" (campos de data e hora). Os campos "Data" e "Hora" contêm os valores "dd/mm/aaaa" e "00:00" respectivamente. Na base da interface, há três botões: "Limpar", "Agendar" e "Voltar". O fundo da interface é azul com formas abstratas.

Fonte: Autoria própria

De forma a auxiliar na padronização das interfaces do sistema e facilitar sua visualização pelos desenvolvedores, foi produzido o Design System, que é um conjunto de elementos, regras e componentes de design utilizados para garantir consistência visual e de interação em um sistema ou produto. Ele é desenvolvido com o objetivo de facilitar o trabalho dos designers e desenvolvedores, fornecendo um conjunto de elementos reutilizáveis e pré-definidos que podem ser aplicados em diferentes partes do sistema. Logo abaixo, na figura 9, pode ser visualizado o Design System para as interfaces do sistema Galenus.

Figura 9: Design System do sistema Galenus



Fonte: Autoria própria

6.2 Descrição do Código

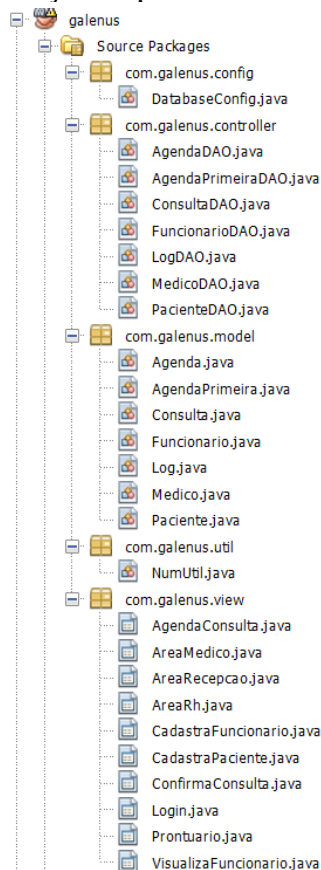
O código do sistema foi desenvolvido utilizando a linguagem orientada a objetos Java e o framework Spring. Essas tecnologias foram escolhidas para lidar com a lógica de negócios, acessar bancos de dados, processar as requisições dos usuários e fornecer os dados e serviços necessários para a interface ou outras partes do sistema.

Além disso, para o desenvolvimento da interface do sistema, foi utilizado a biblioteca gráfica JavaFX, em conjunto com a IDE NetBeans. O JavaFX é uma biblioteca de interface de usuário que permite criar interfaces gráficas ricas e interativas para aplicativos Java, já o NetBeans é uma IDE popular que oferece suporte para o desenvolvimento de aplicativos JavaFX, além de oferecer recursos de edição, depuração e gerenciamento de projetos.

Para a organização de arquivos, pacotes, classes ou quaisquer estruturas utilizadas no desenvolvimento do projeto, foi empregado a arquitetura MVC (Model-View-Controller) onde foi organizada da seguinte maneira:

- Pacote "model": Este pacote contém as classes que representam os objetos principais do sistema, como "Agenda", "Consulta", "Funcionário", entre outros. Essas classes possuem os atributos e métodos relacionados às suas respectivas entidades.
- Pacote "view": Neste pacote, encontram-se as classes responsáveis pela interface do usuário, como "AgendaConsulta", "AreaRH", "Login" e entre outras classes que lidam com a interação do usuário.
- Pacote "controller": Neste pacote, encontram-se as classes responsáveis por lidar com a persistência dos dados do sistema e que controla o fluxo da aplicação. Temos como exemplos as classes "AgendaDAO", "ConsultaDAO" e "MedicoDAO".
- Pacote "util": Aqui contém classes utilitárias que oferecem funcionalidades auxiliares para o sistema. Por exemplo, a classe "NumUtil" do sistema pode fornecer métodos para formatação e manipulação de datas.
- Pacote "config": Esse pacote geralmente contém classes responsáveis por definir e carregar configurações, estabelecer conexões com bancos de dados, configurar bibliotecas e outros componentes necessários para o funcionamento adequado do sistema. No sistema Galenus, foi definido a classe "DatabaseConfig", que realiza as configurações do banco de dados.

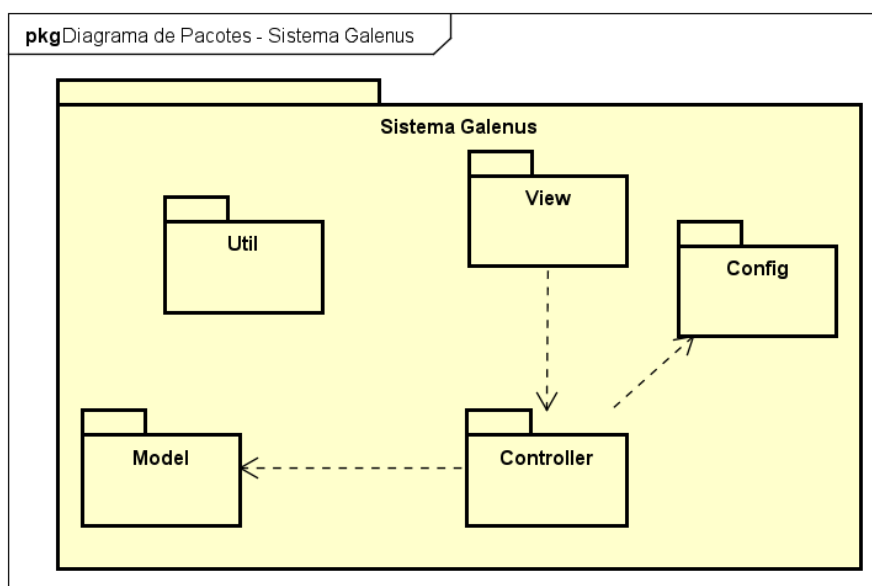
Figura 10: Organização de pacotes e classes do projeto - NetBeans



Fonte: Autoria própria

Além disso, também foi desenvolvido o diagrama de pacotes, por meio do qual é possível representar os pacotes e as dependências entre eles.

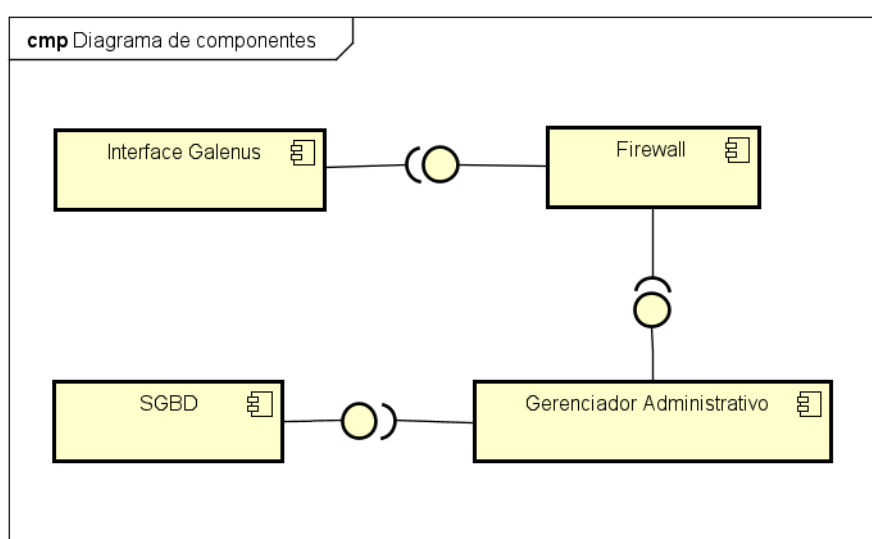
Figura 11: Diagrama de pacotes do sistema



Fonte: Autoria própria

De forma a complementar a representação da estrutura do sistema, também foi desenvolvido o diagrama de componentes que demonstra a estrutura interna e as dependências dos componentes de um sistema. Ele mostra como os diferentes componentes de um sistema se relacionam uns com os outros e como eles interagem para fornecer funcionalidade ao sistema.

Figura 12: Diagrama de componentes do sistema



Fonte: Autoria própria

O diagrama demonstrado na figura 12, representa os seguintes componentes:

- **Interface Galenus:** Esse componente representa simplesmente a interface do sistema Galenus, por meio do qual o usuário pode solicitar serviços como login, agendamento de consultas e cadastro de pacientes. Esse componente apresenta uma interface requerida com o componente firewall, uma vez que ele solicita ao firewall a transmissão dos eventos ocorridos sobre a interface ao Gerenciador Administrativo.
- **Firewall:** Esse componente representa um software que implementa um sistema de firewall, ou seja, é um software de segurança que tenta impedir que usuários não autorizados invadam o sistema. Esse componente não faz realmente parte do sistema, entretanto, por medidas de segurança, todo o tráfego externo deve passar por ele antes de se comunicar com o sistema propriamente dito.
- **Gerenciador Administrativo:** Esse componente representa o módulo do sistema responsável por gerenciar as solicitações realizadas pelos funcionários da clínica na interface do sistema Galenus. Ele deve interpretar os eventos ocorridos na interface e solicitar as execuções adequadas a eles.
- **SGBD:** Representa o Sistema Gerenciador de Banco de Dados responsável por persistir e recuperar os dados do sistema.

Por fim, em relação a nomenclatura de classes, objetos e as funções do código, utilizamos a nomeação das classes usando a convenção CamelCase, começando com uma letra maiúscula. Os objetos e variáveis são nomeados usando a convenção camelCase, começando com uma letra minúscula. As funções e métodos são nomeados usando a convenção camelCase, começando com uma letra minúscula.

7 Considerações Finais

Neste projeto, o sistema de gerenciamento hospitalar Galenus foi desenvolvido com o objetivo de otimizar a gestão administrativa da clínica médica Galenus. O sistema foi implementado de forma a proporcionar funcionalidades essenciais para o manuseio e registro de informações, eliminando a necessidade de atividades manuais e recursos físicos. Os resultados obtidos foram altamente satisfatórios, atendendo às expectativas e os requisitos de todos os envolvidos.

O sistema Galenus trouxe benefícios e melhorias significativas ao ambiente hospitalar, aumentando a eficiência operacional por meio de um agendamento de consultas ágil e preciso, resultando em uma melhor utilização dos recursos disponíveis. Outro aspecto positivo é o acesso rápido e fácil aos prontuários dos pacientes possibilitando em um atendimento mais preciso e personalizado, melhorando a qualidade do serviço oferecido. Além disso, a substituição dos registros físicos por registros eletrônicos também contribuiu para a redução de erros e retrabalho, minimizando falhas de transcrição e perda de informações.

Desse modo, o sistema com o conjunto de funcionalidades oferecidas, demonstrou-se altamente eficiente e vantajoso para a clínica médica. Através do Galenus, a gestão administrativa alcançará um novo patamar de qualidade, possibilitando um melhor controle das informações dos funcionários, pacientes e consultas, resultando em uma tomada de decisão mais embasada, maior agilidade no agendamento e reagendamento de consultas, e um atendimento mais personalizado aos pacientes.

Por outro lado, em relação ao processo de criação do projeto, pode-se afirmar que o ponto forte se teve com a documentação bem estruturada do projeto, incluindo a diagramação e os requisitos bem definidos, contribuindo para um melhor entendimento do escopo do sistema e facilitando sua implementação. Isso resultou em uma maior visibilidade do projeto final e possibilitou que seu desenvolvimento tivesse um bom rendimento durante todas as fases de construção, além disso, permitiu também uma visão para as futuras melhorias e evoluções do sistema.

Durante toda a concepção do projeto, não foram identificados muitos impedimentos, o que permitiu uma implementação tranquila e bem-sucedida do sistema Galenus, no entanto, é importante destacar que existem áreas de aprimoramento a serem exploradas no futuro, como a expansão de funcionalidades, incluindo recursos de telemedicina para consultas virtuais e monitoramento remoto dos pacientes. Além disso, outra possível melhoria seria a interface do usuário, que pode ser otimizada para uma experiência mais intuitiva e amigável.

Em conclusão, pode-se dizer que o desenvolvimento do projeto foi executado com sucesso, atendendo todas as expectativas dos envolvidos e entregando todas as funcionalidades com êxito. O sistema com certeza irá representar um avanço significativo para a clínica médica Galenus, proporcionando maior eficiência operacional, redução de erros e melhorias na qualidade do atendimento. Com resultados satisfatórios alcançados e perspectivas de expansão no futuro próximo, o Galenus se mostra uma solução promissora para o gerenciamento administrativo de clínicas médicas.

8 Bibliografia

LEITE, Cesar Eduardo. LAURETTI, Rosa Maria de Souza. A Operacionalização de Sistemas de Informação na Gestão Hospitalar: um Estudo de Caso de Dois Pequenos Hospitais. 2011.

Visão Geral do PDS – Scrum. Tribunal Regional do Trabalho do Paraná. Disponível em: <<https://www.trt9.jus.br/pds/index.htm>>. Acesso em: 22 de mar. de 2023.

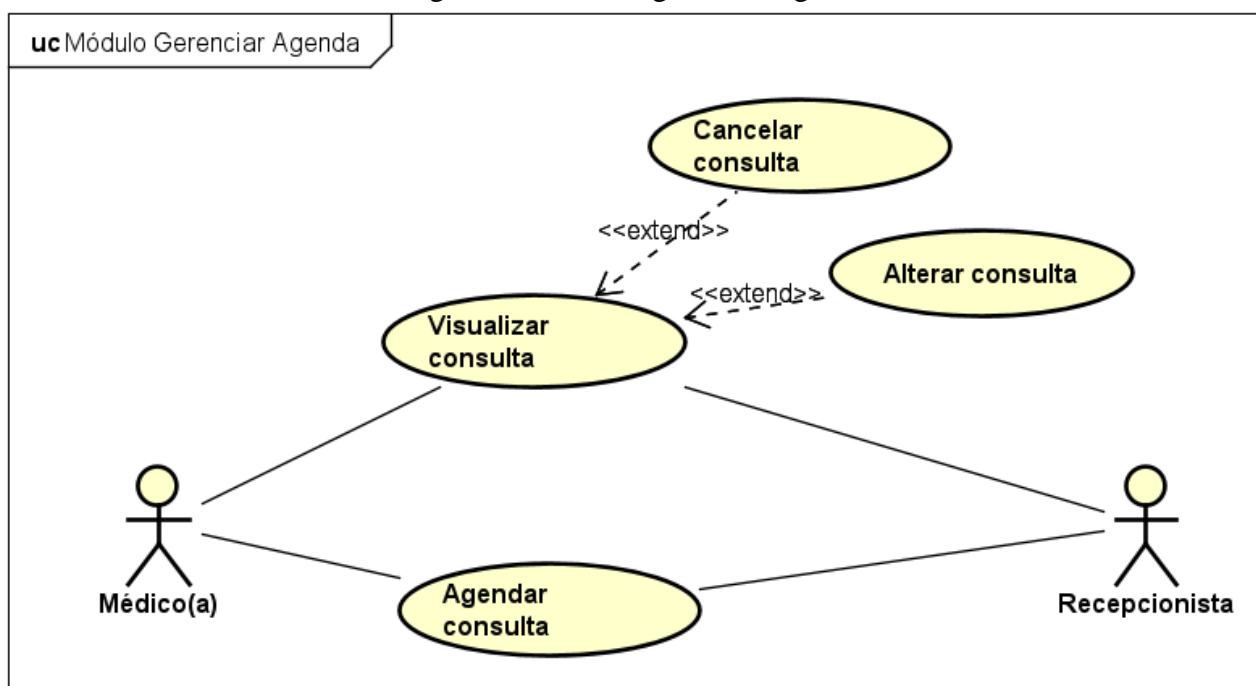
Scrum.org. O Guia do Scrum. Disponível em: <<https://www.scrum.org/>>. Acesso em: 15 jun. 2023.

APÊNDICES

Apêndice A: Módulos dos casos de uso do sistema Galenus	38
Apêndice B: Diagramas de atividades uso do sistema Galenus	41
Apêndice C: Protótipos das interfaces do sistema Galenus	45

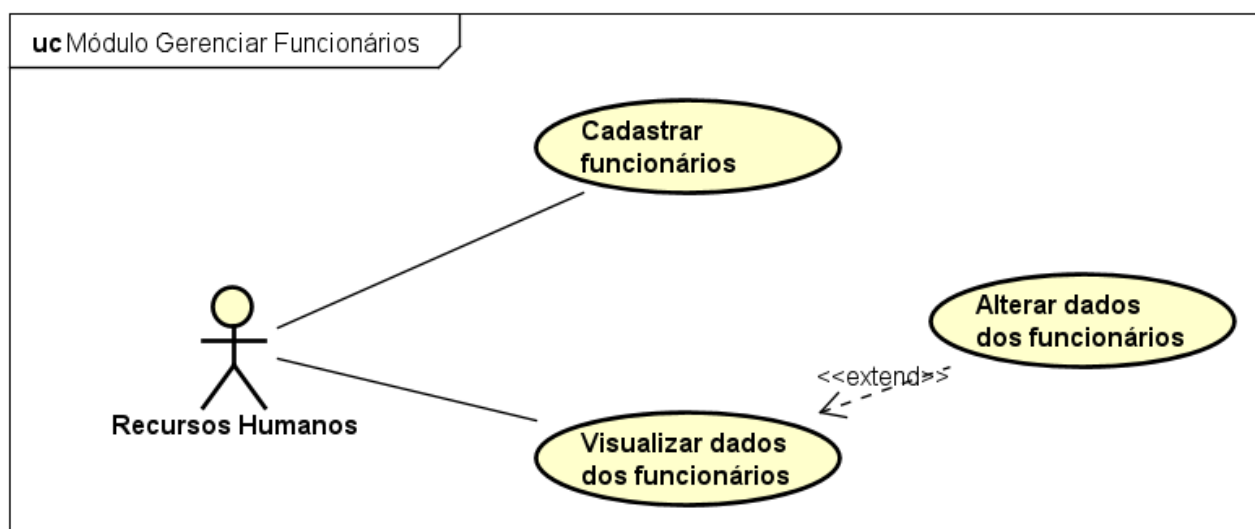
APÊNDICE A - Módulos dos casos de uso do sistema Galenus

Figura 13: Módulo gerenciar agenda



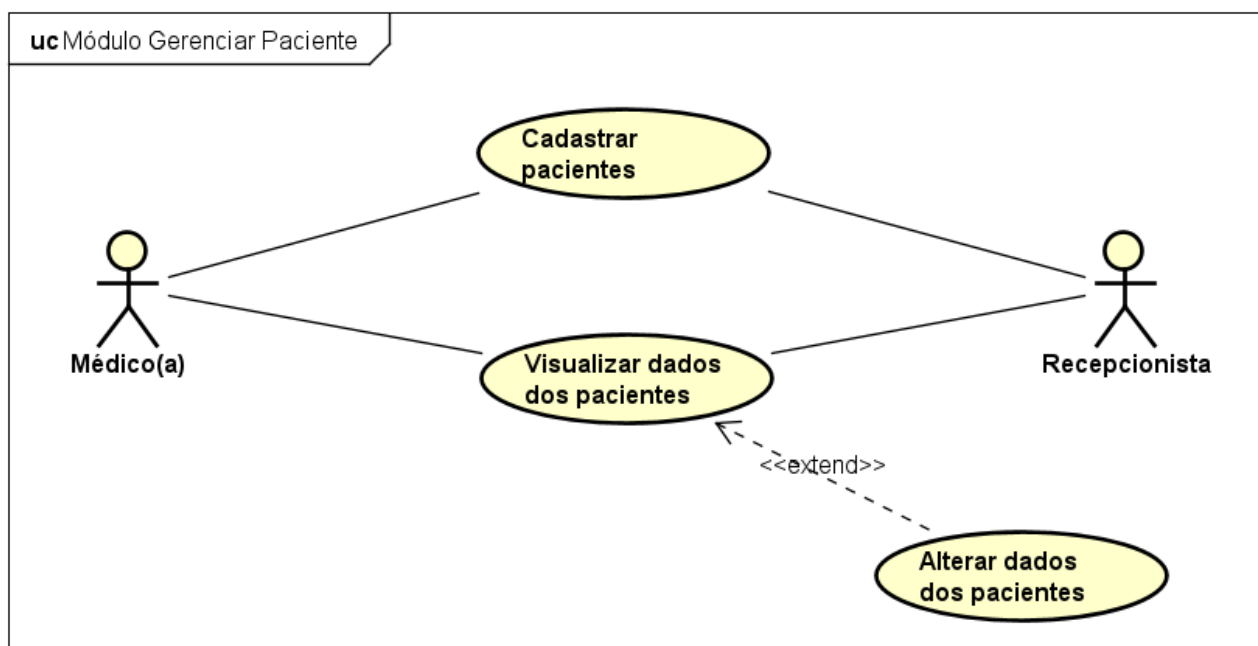
Fonte: Autoria própria

Figura 14: Módulo gerenciar funcionários



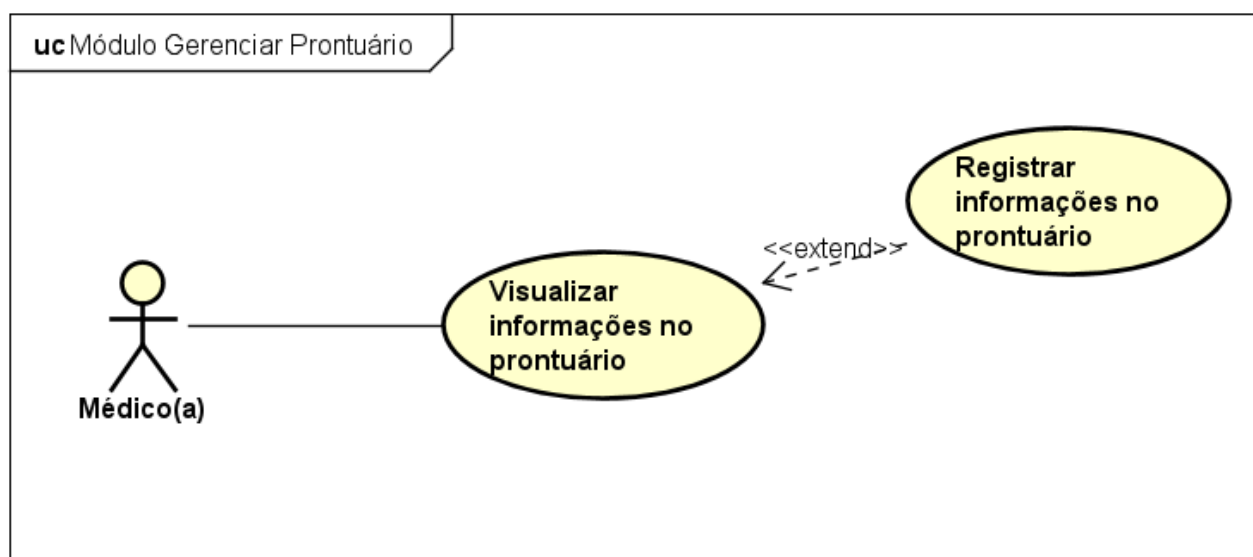
Fonte: Autoria própria

Figura 15: Módulo gerenciar pacientes



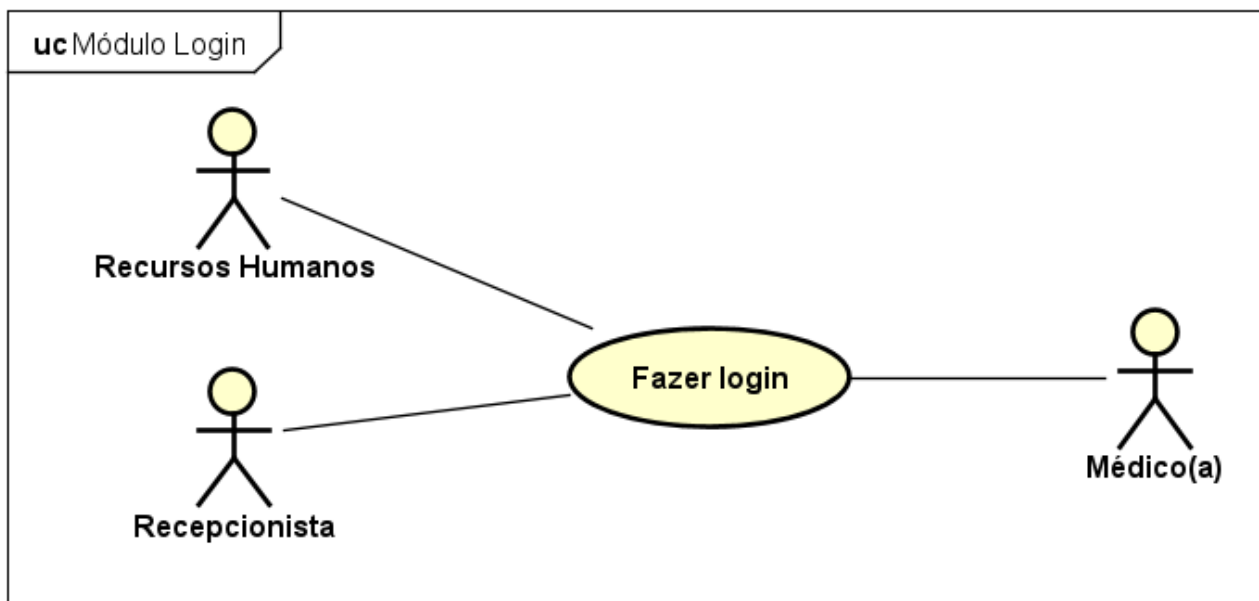
Fonte: Autoria própria

Figura 16: Módulo gerenciar prontuário



Fonte: Autoria própria

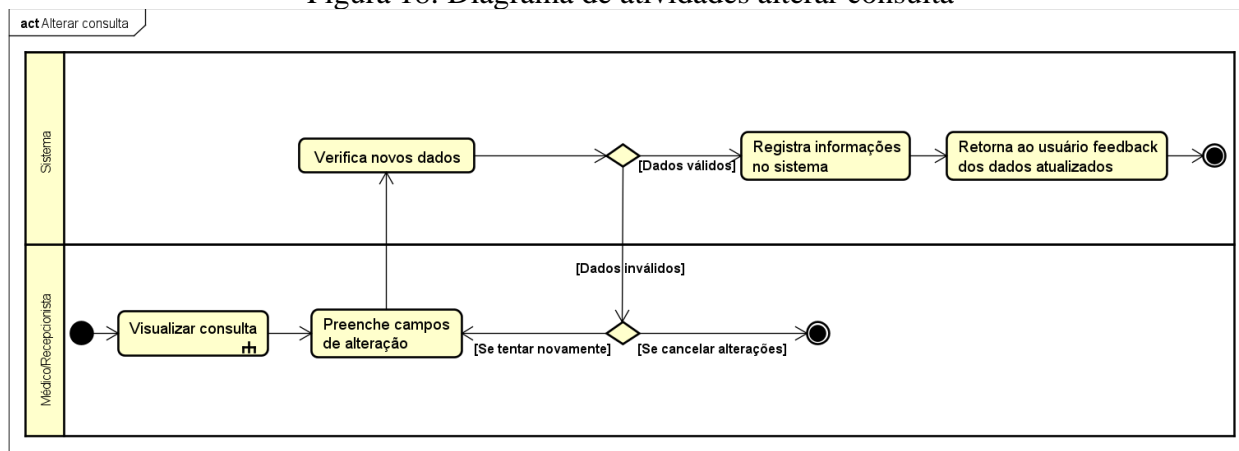
Figura 17: Módulo fazer login



Fonte: Autoria própria

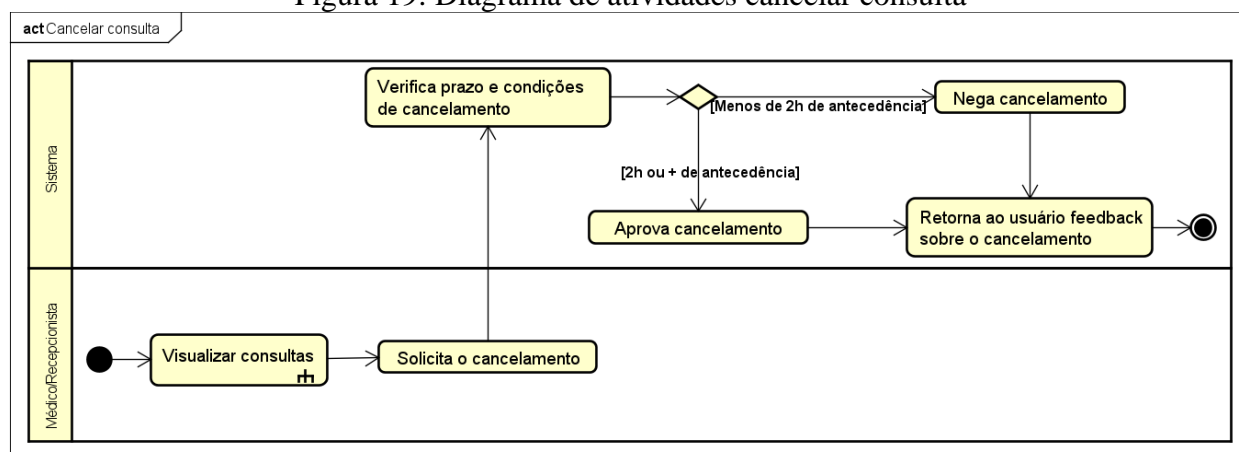
APÊNDICE B - Diagramas de atividades uso do sistema Galenus

Figura 18: Diagrama de atividades alterar consulta



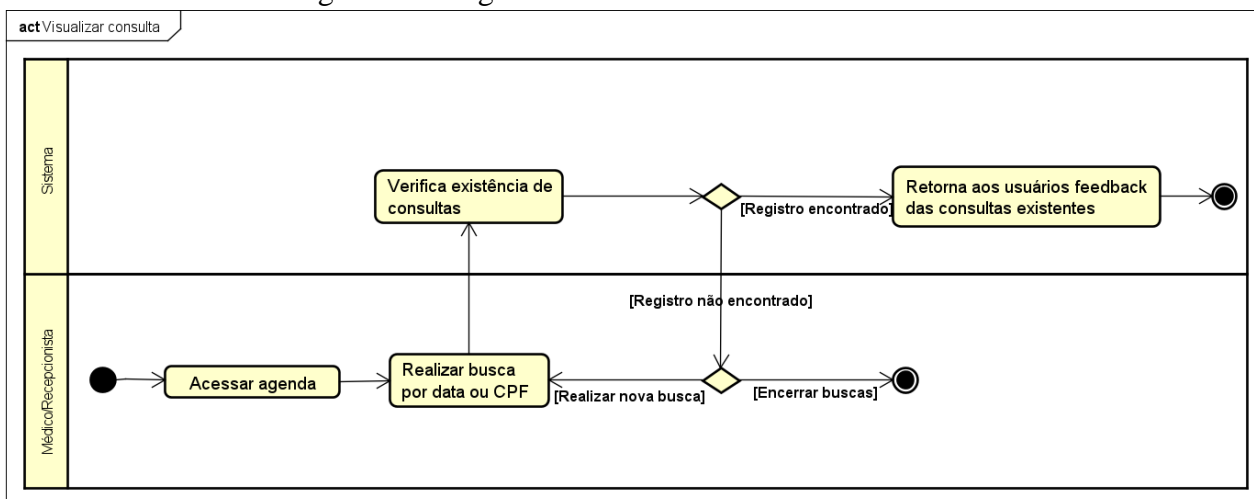
Fonte: Autoria própria

Figura 19: Diagrama de atividades cancelar consulta



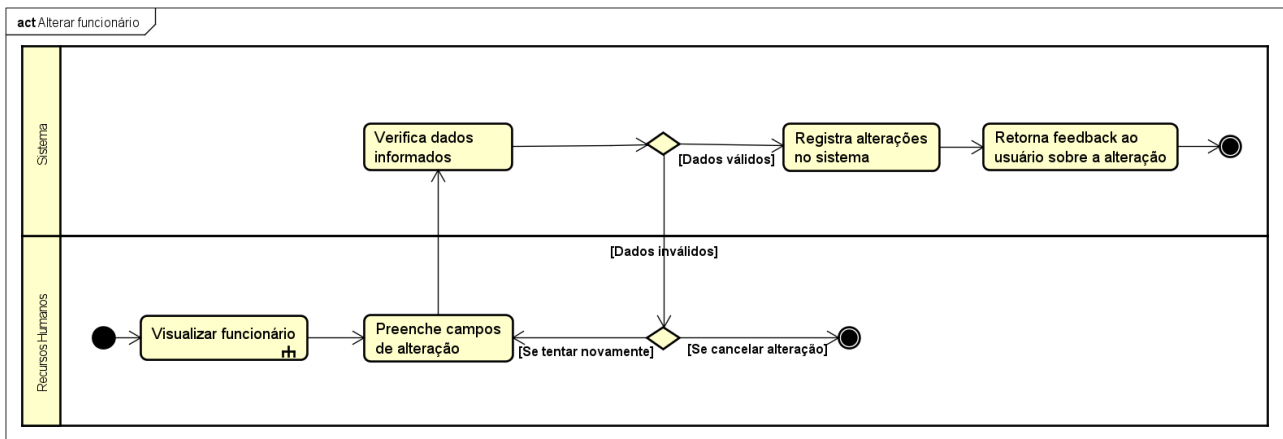
Fonte: Autoria própria

Figura 20: Diagrama de atividades visualizar consulta



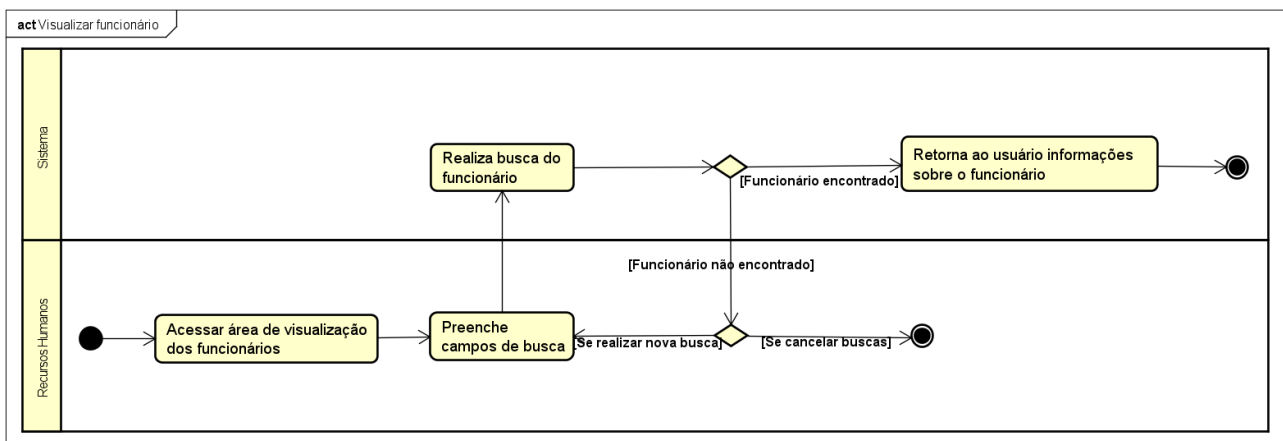
Fonte: Autoria própria

Figura 21: Diagrama de atividades alterar funcionário



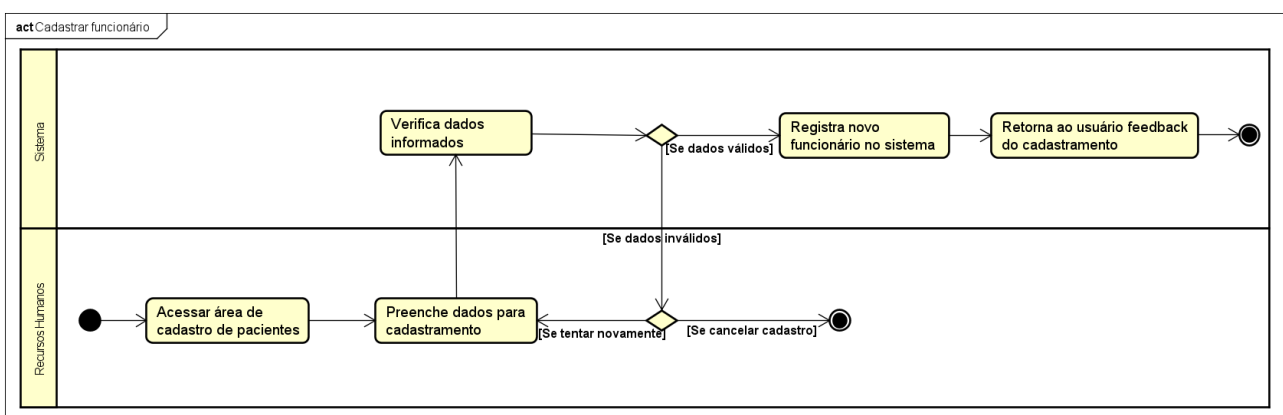
Fonte: Autoria própria

Figura 22: Diagrama de atividades visualizar funcionário



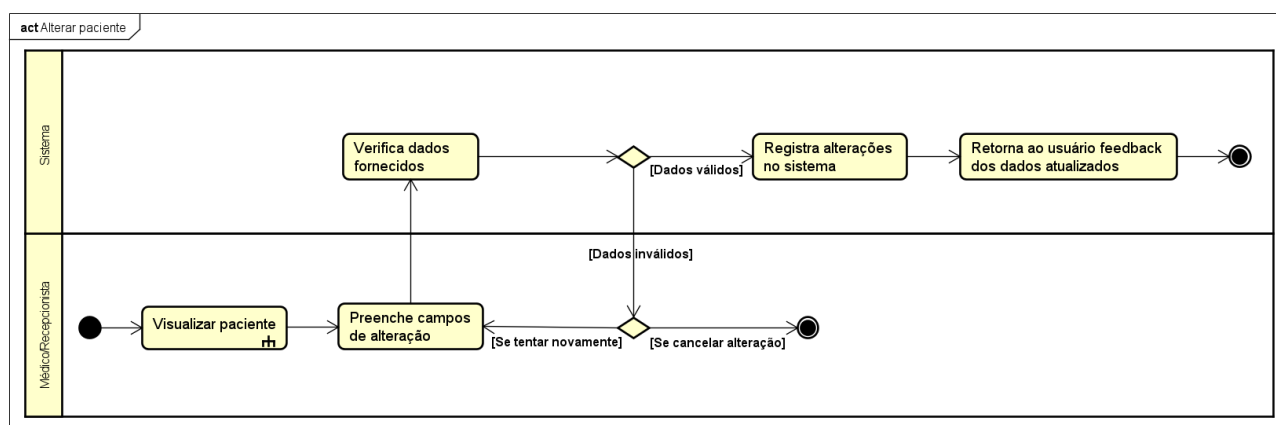
Fonte: Autoria própria

Figura 23: Diagrama de atividades cadastrar funcionários



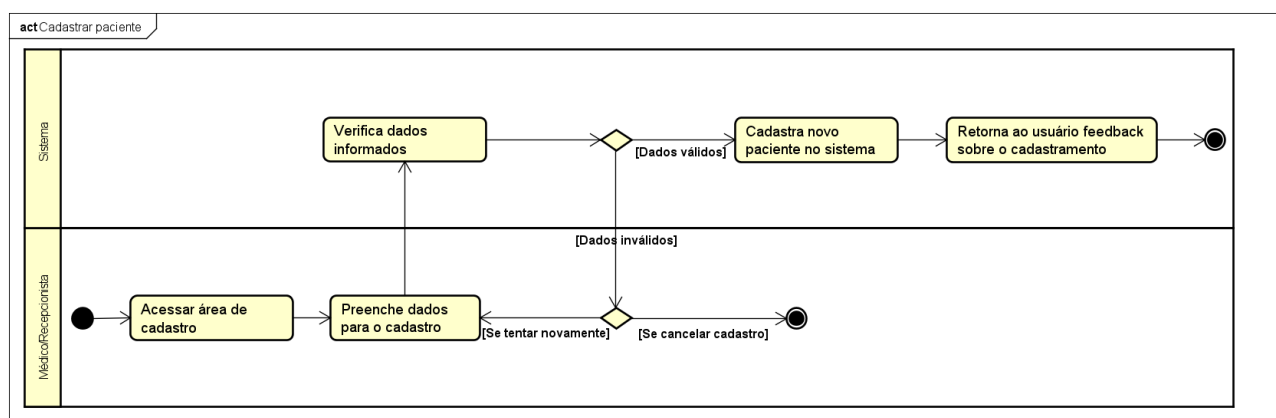
Fonte: Autoria própria

Figura 24: Diagrama de atividades alterar pacientes



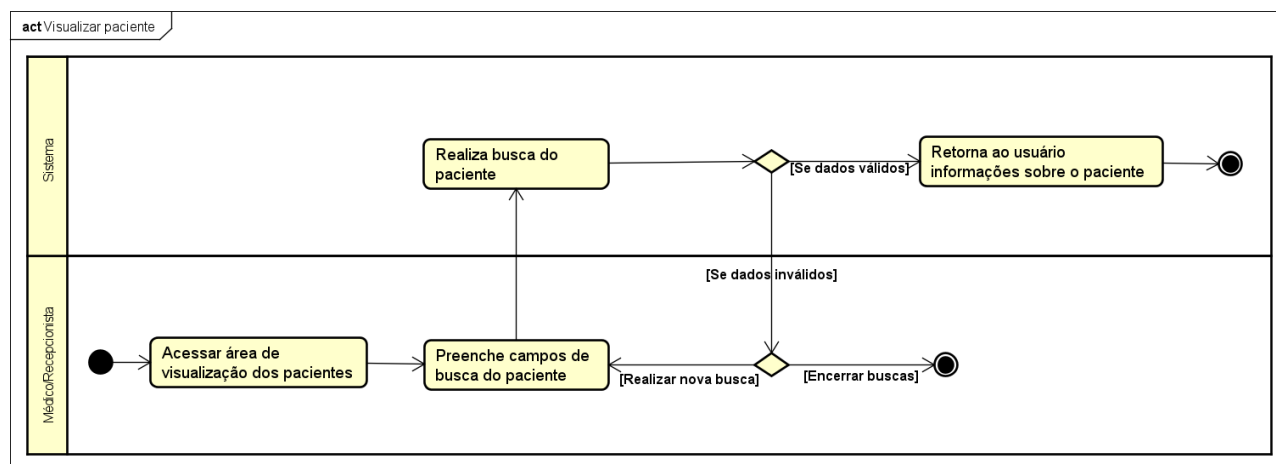
Fonte: Autoria própria

Figura 25: Diagrama de atividades cadastrar pacientes



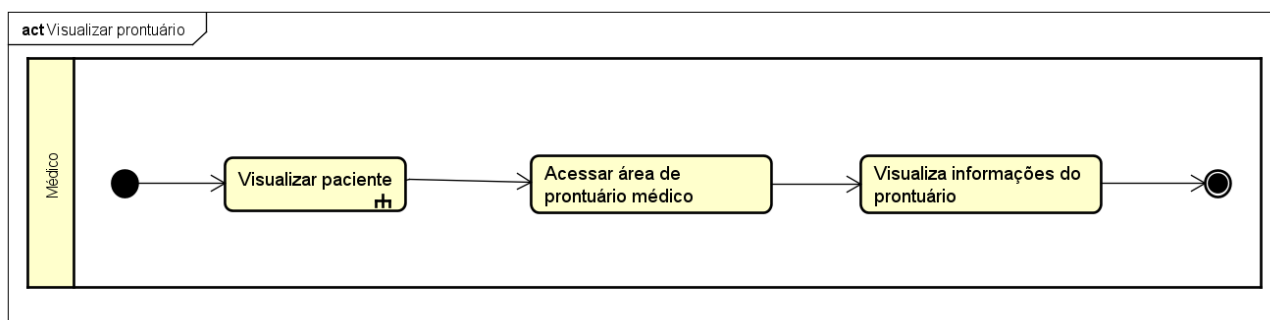
Fonte: Autoria própria

Figura 26: Diagrama de atividades visualizar pacientes



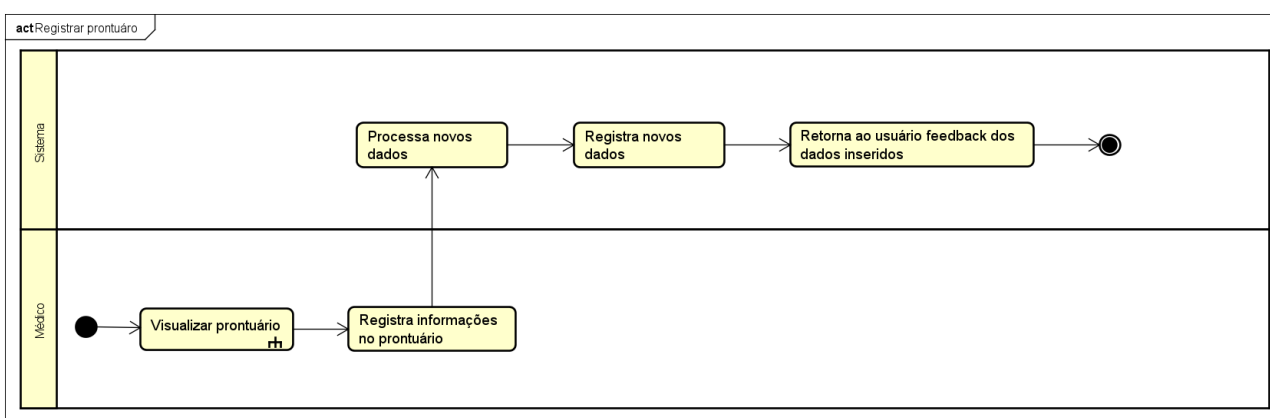
Fonte: Autoria própria

Figura 27: Diagrama de atividades visualizar prontuário



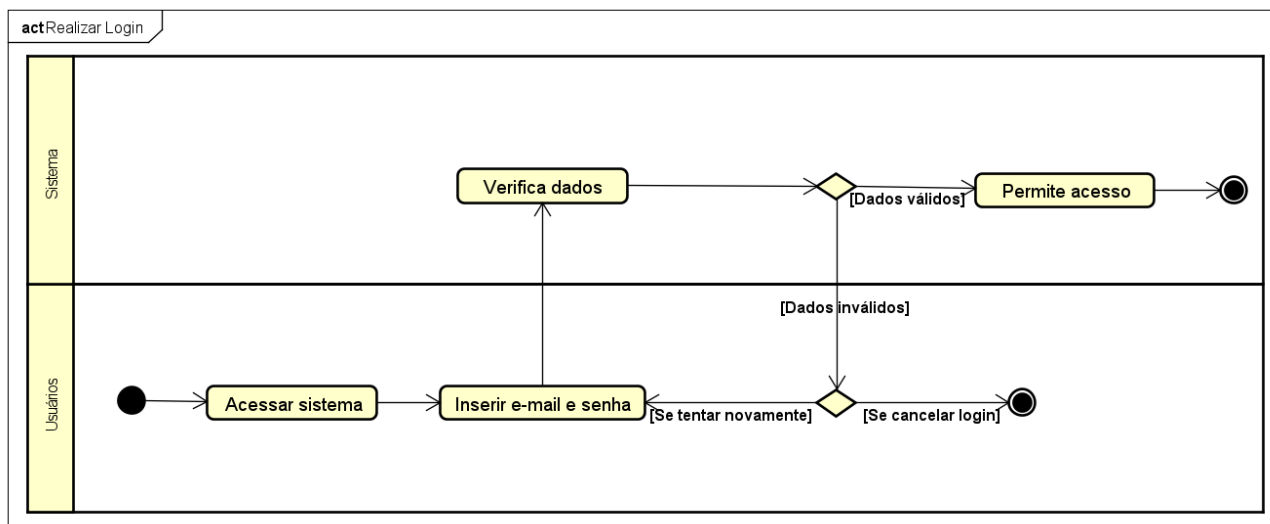
Fonte: Autoria própria

Figura 28: Diagrama de atividades registrar no prontuário



Fonte: Autoria própria

Figura 29: Diagrama de atividades realizar login



Fonte: Autoria própria

APÊNDICE C - Protótipos das interfaces do sistema Galenus

Figura 30: Tela principal para os funcionários dos Recursos Humanos (RH)




Fonte: Autoria própria

Figura 31: Tela de cadastro de funcionários



Fonte: Autoria própria

Figura 32: Tela cadastrar médico



The image shows a web form titled "Cadastrar Médico" (Register Doctor) on a blue background with abstract purple and blue shapes. The form consists of two columns of input fields. The first column contains fields for "Nome" (Name), "CPF" (CPF), "Data de nascimento" (Date of birth), and "Endereço" (Address). The second column contains fields for "Telefone" (Phone), "Especialidade" (Specialty), "CRM", and "Sexo" (Sex). Below the input fields is a "Cadastrar" (Register) button. At the bottom right, there are two buttons: "Limpar" (Clear) and "Voltar" (Back).

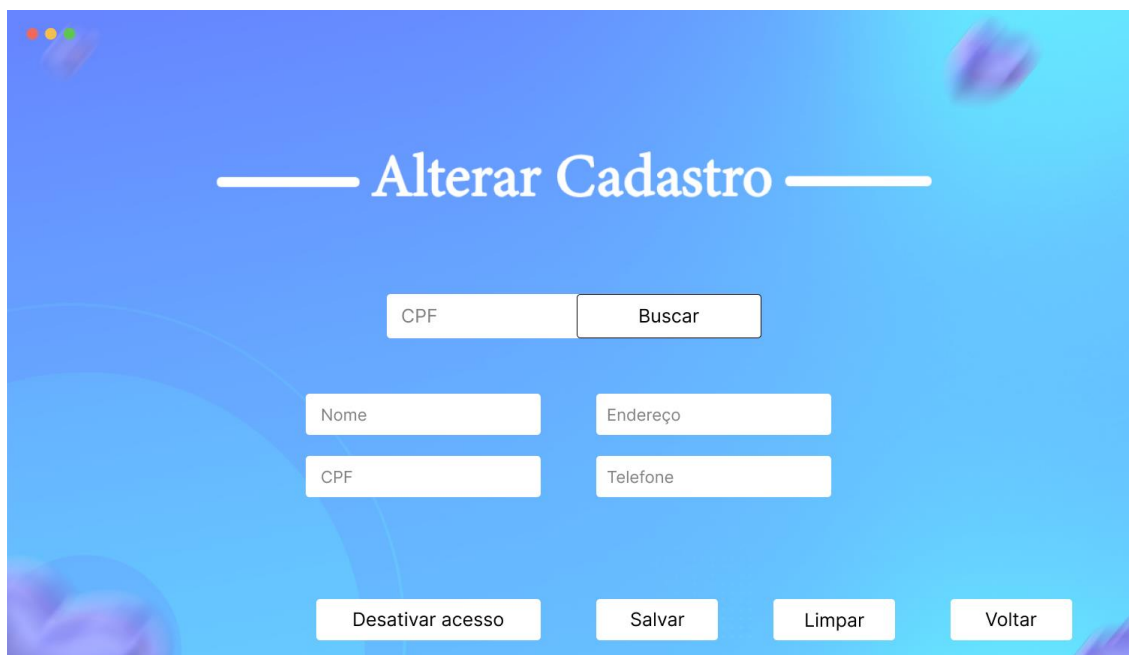
Nome	Telefone
CPF	Especialidade
Data de nascimento	CRM
Endereço	Sexo

Cadastrar

Limpar Voltar

Fonte: Autoria própria

Figura 33: Tela alterar cadastro



The image shows a web form titled "Alterar Cadastro" (Edit Record) on a blue background with abstract purple and blue shapes. The form has a search section at the top with a "CPF" input field and a "Buscar" (Search) button. Below this are four input fields arranged in two rows: "Nome" (Name) and "Endereço" (Address) in the first row, and "CPF" and "Telefone" (Phone) in the second row. At the bottom, there are four buttons: "Desativar acesso" (Deactivate access), "Salvar" (Save), "Limpar" (Clear), and "Voltar" (Back).

CPF	Buscar
Nome	Endereço
CPF	Telefone

Desativar acesso Salvar Limpar Voltar

Fonte: Autoria própria

Figura 34: Tela visualizar cadastro



A screenshot of a web application titled "Visualizar Cadastro". The background is a light blue gradient with abstract purple and blue shapes. The title is centered at the top in a large, white, serif font, flanked by horizontal lines. Below the title, there are two columns of input fields. The left column contains fields for "Nome", "CPF", "Data de nascimento", and "CRM". The right column contains fields for "Endereço", "Telefone", "Sexo", and "Especialidade". Below these fields, there is a single "CPF" field, followed by a "Buscar" button. At the bottom right, there are two buttons: "Limpar" and "Voltar".

Visualizar Cadastro

Nome

Endereço

CPF

Telefone

Data de nascimento

Sexo

CRM

Especialidade

CPF

Buscar

Limpar

Voltar

Fonte: Autoria própria

Figura 35: Tela principal da recepção



A screenshot of a web application titled "Recepção". The background is a light blue gradient with abstract purple and blue shapes. The title is centered at the top in a large, white, serif font, flanked by horizontal lines. Below the title, there are five buttons arranged in two rows. The first row contains "Agendar Consulta" and "Visualizar Consulta". The second row contains "Confirmar Consulta" and "Cancelar Consulta". Below these, there is a single button "Cadastrar paciente". At the bottom right, there is a button "Sair".

Recepção

Agendar Consulta

Visualizar Consulta

Confirmar Consulta

Cancelar Consulta

Cadastrar paciente

Sair

Fonte: Autoria própria

Figura 36: Tela confirmar consulta

A interface de usuário para confirmar uma consulta. O título "Confirmar Consulta" está centralizado no topo. Abaixo, há campos de entrada para Nome, Telefone, CPF, Especialidade, Médico e Forma de pagamento. Na base, há campos para Data (formato dd/mm/aaaa) e Hora (formato 00:00). Botões para "Cofirmar" e "Voltar" estão localizados no canto inferior direito.

— Confirmar Consulta —

Nome Telefone

CPF Especialidade

Médico Forma de pagamento ▾

Data dd/mm/aaaa Hora 00:00

Cofirmar Voltar

Fonte: Autoria própria

Figura 37: Tela agendar consulta

A interface de usuário para agendar uma consulta. O título "Agendar Consulta" está centralizado no topo. Abaixo, há campos de entrada para Nome, Telefone, Médico (menu suspenso) e Especialidade (menu suspenso). Na base, há campos para Data (formato dd/mm/aaaa) e Hora (formato 00:00). Botões para "Limpar", "Agendar" e "Voltar" estão localizados no canto inferior direito.

— Agendar Consulta —

Nome Telefone

Médico ▾ Especialidade ▾

Data dd/mm/aaaa Hora 00:00

Limpar Agendar Voltar

Fonte: Autoria própria

Figura 38: Tela cadastrar pacientes

A interface de usuário para cadastrar um paciente. O título "Cadastrar Paciente" está centralizado no topo. Abaixo dele, há seis campos de entrada organizados em duas colunas: Nome, Telefone, CPF, Sexo, Data de nascimento e Endereço. Um botão "Cadastrar" está centralizado abaixo dos campos. No canto inferior direito, há dois botões: "Limpar" e "Voltar".

— Cadastrar Paciente —

Nome Telefone

CPF Sexo

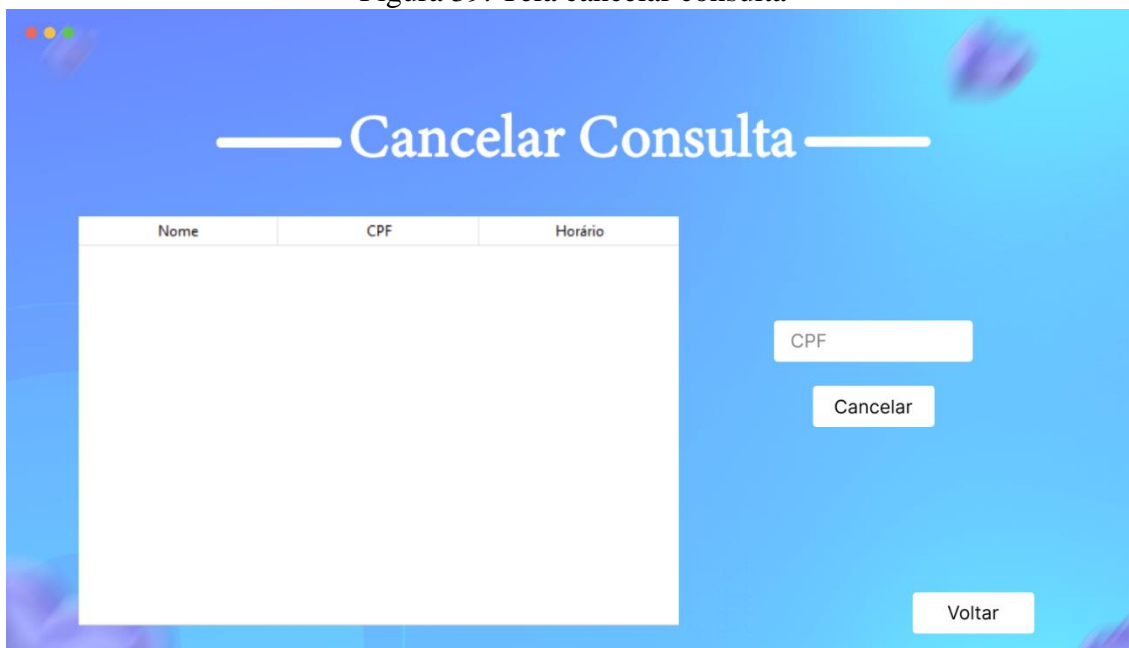
Data de nascimento Endereço

Cadastrar

Limpar Voltar

Fonte: Autoria própria

Figura 39: Tela cancelar consulta

A interface de usuário para cancelar uma consulta. O título "Cancelar Consulta" está centralizado no topo. Abaixo dele, há uma tabela com três colunas: Nome, CPF e Horário. À direita da tabela, há um campo de entrada "CPF" e um botão "Cancelar". No canto inferior direito, há um botão "Voltar".

— Cancelar Consulta —

Nome	CPF	Horário
------	-----	---------

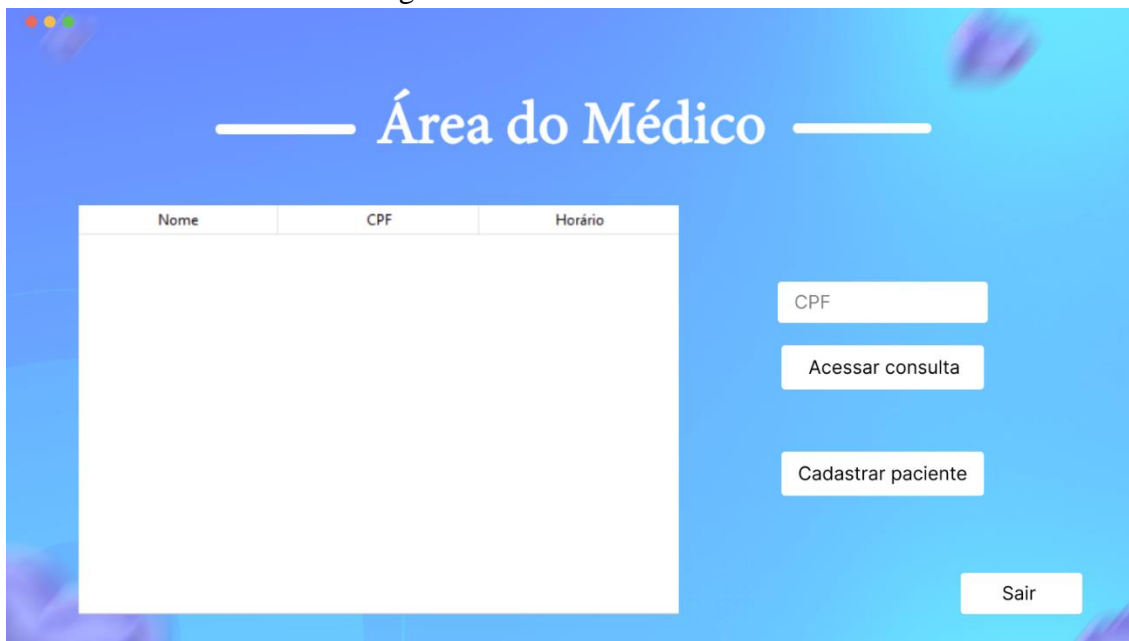
CPF

Cancelar

Voltar

Fonte: Autoria própria

Figura 40: Tela área do médico

A interface da tela 'Área do Médico' possui um fundo azul com uma abstração de corais. No topo, o título 'Área do Médico' é centralizado entre duas linhas brancas. Abaixo, há uma tabela com três colunas: 'Nome', 'CPF' e 'Horário'. À direita da tabela, há um campo de entrada para 'CPF', seguido por dois botões: 'Acessar consulta' e 'Cadastrar paciente'. No canto inferior direito, há um botão 'Sair'.

Nome	CPF	Horário
------	-----	---------

CPF

Acessar consulta

Cadastrar paciente

Sair

Fonte: Autoria própria

Figura 41: Tela do prontuário do paciente

A interface da tela 'Prontuário' possui o mesmo fundo azul com abstração de corais. O título 'Prontuário' é centralizado no topo entre duas linhas brancas. À esquerda, há uma coluna vertical com seis campos de entrada: 'Nome', 'Data de nascimento', 'Sexo', 'Queixa principal', 'Hipótese de diagnóstico' e 'Diagnóstico'. À direita, há uma grade de três colunas e duas linhas de campos de entrada: 'Doenças pré-existentes', 'História Clínica', 'Tratamento' e 'Conduta'. Na base, há três botões: 'Limpar', 'Salvar' e 'Voltar'.

Nome

Data de nascimento

Sexo

Queixa principal

Hipótese de diagnóstico

Diagnóstico

Doenças pré-existentes

História Clínica

Tratamento

Conduta

Limpar

Salvar

Voltar

Fonte: Autoria própria