

# + SAÚDE

## Detalhamento dos Casos de Uso

### Histórico de Revisões

Data	Versão	Descrição	Autor
20/09/2023	0.1	Escrita dos casos de uso	Luana Schelb
21/09/2023	0.1	Escrita dos casos de uso	Elizângela
09/11/2023	0.2	Adicionado diagrama de classes	Luana Schelb
10/11/2023	0.3	Adicionados detalhes nos casos de uso	Luana Schelb
10/11/2023	0.4	Adicionado detalhes aos casos de uso	Caio Capua
15/12/2023	0,5	Adicionado diagrama de sequencia	Caio Capua
15/12/2023	0,6	Adicionado Diagrama de classes	Caio Capua
15/12/2023	0,7	Mapeamento de classes para codigo java	Caio Capua

### Definição do problema

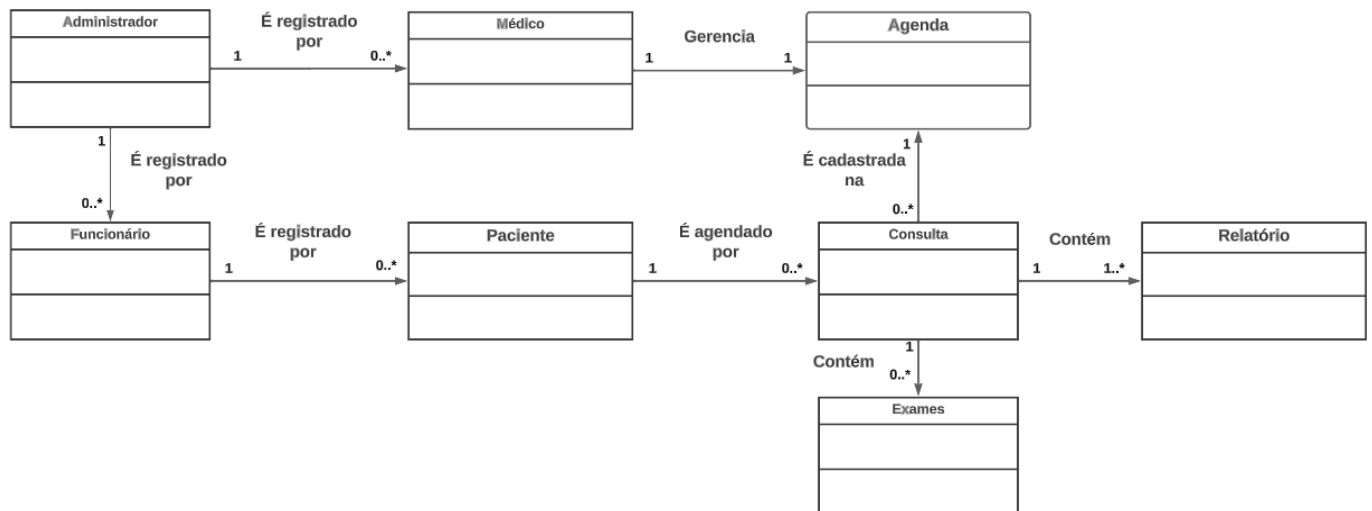
Muitas vezes é necessário ligar para o consultório ou ir ao local para marcar consultas. Outras vezes, o mesmo acontece quando precisamos pegar os resultados de exames solicitados pelo médico. Dessa forma, perde-se um certo tempo realizando essas atividades devido a possibilidade de ter filas, visto que a maioria das pessoas fazem isso em horários específicos.

- Plano de Desenvolvimento

ID	Nome do Caso de Uso	Descrição Resumida	Importância	Versão
CU-01	Cadastrar funcionários	O administrador realiza o cadastro dos funcionários no sistema	Alta	0.1
CU-02	Cadastrar médicos	O administrador realiza o cadastro dos médicos no sistema	Alta	0.1
CU-03	Cadastrar paciente	O funcionário realiza o cadastro dos pacientes no sistema	Alta	0.1
CU-04	Preencher relatório da consulta	O médico preenche um relatório resumindo a consulta do paciente.	Alta	0.1
CU-05	Buscar Médicos Credenciados pelo	O paciente busca em sua aplicação	Alta	0.1

	Paciente	os médicos credenciados		
CU-06	Agendar Consulta pelo Paciente	O paciente realiza o agendamento de uma consulta na sua aplicação	Alta	0.1
CU-07	Visualizar dados da consulta pelo Paciente	O paciente visualiza suas consultas e seus dados como data, médicos, estado da consulta (agendada, finalizada, exames prontos, etc.)	Alta	0.1
CU-08	Consultar agenda de médicos	O paciente consegue consultar o histórico de suas consultas	Alta	0.1
CU-09	Consultar resultado de exames	O paciente consegue visualizar o resultado dos exames solicitados nas consultas	Alta	0.1
CU-10	Bloquear Agenda	O médico consegue bloquear o(s) dia(s) e/ou horário(s) em que não irá realizar consultas	Alta	0.1
CU-11	Cancelar Consulta	O paciente consegue realizar o cancelamento de agendamento da consulta	Alta	0.1

- Diagrama de casos de uso



- Especificação Dos casos de uso

- [CU-01] Cadastrar Funcionários

- Descrição Resumida dos Atores

Administrador do sistema – onde o Administrador é um usuário administrador com todas permissões do sistema

- Pré-condições

Administrador deve estar autenticado

- Cenário Principal

- O Administrador clica no botão “Cadastrar funcionário” no painel administrativo
- O sistema carrega uma página com os seguintes componentes
  - Um formulário contendo os campos:
    - Nome do funcionário
    - Função
  - O Administrador preenche os campos obrigatórios com as informações do funcionário que deseja cadastrar.
  - O Administrador pode, opcionalmente, preencher campos adicionais, como:
    - Data de Nascimento
    - Endereço
    - Fotografia do funcionário
  - O Administrador verifica se todas as informações estão corretas e completas.
  - Após verificar, o Administrador clica no botão “Cadastrar” e confirmando o cadastro do funcionário.

- Cenários Alternativos

- Caso o administrador encontre erros o funcionário confirma as informações e altera os dados ou cancela o cadastro

- Pós-condições
  - Funcionário deve ser cadastrado no sistema
- Requisitos Especiais
  - Segurança:

Garantir que o sistema tenha mecanismos robustos de autenticação para o administrador, dado que a pré-condição é que ele esteja autenticado.

Implementar permissões de acesso para garantir que apenas administradores tenham acesso ao caso de uso "Cadastrar Funcionários".

- Armazenamento de Dados:

Especificar requisitos relacionados ao armazenamento seguro e eficiente das informações do funcionário no banco de dados.

- Informações Adicionais
  - Políticas de Senhas

Se aplicável, incluir informações sobre políticas de senhas para proteger o acesso do administrador ao sistema.

- Campos Obrigatorio e Opcionais:

Esclarecer quais campos são obrigatórios e quais são opcionais durante o cadastro do funcionário.

- Protótipo Essencial
  - Telas Principais:

Identificar as telas essenciais que fazem parte do protótipo, destacando aquelas que são críticas para a compreensão do fluxo de cadastro de funcionários.

- FeedBack ao usuario:

Incluir elementos de feedback ao usuário, como mensagens de sucesso, alertas de erro e confirmações.

- [CU-02] Cadastrar médicos
- Descrição Resumida dos Atores

Administrador do sistema – onde o Administrador é um usuário administrador com todas as permissões do sistema

- Pré-condições

Administrador deve estar autenticado

- Cenário Principal

- O Administrador clica no botão “Cadastrar médico” no painel administrativo
- O sistema carrega uma página com os seguintes componentes
  - Um formulário contendo os campos:
    - Nome do Médico
    - Especialidade
    - CRM (Conselho Regional de Medicina)
    - Endereço de E-mail
    - Telefone de Contato
  - O Administrador preenche os campos obrigatórios com as informações do médico que deseja cadastrar.
  - O Administrador pode, opcionalmente, preencher campos adicionais, como:
    - Data de Nascimento
    - Endereço
    - Fotografia do Médico
    - Horário de Atendimento
  - O Administrador verifica se todas as informações estão corretas e completas.
  - Após verificar, o Administrador clica no botão “Cadastrar” e confirmando o cadastro do médico.

- Cenários Alternativos

- Caso o administrador encontre erros o médico confirma as informações e altera os dados ou cancela o cadastro

- Pós-condições

- Médico deve ser cadastrado no sistema

- Requisitos Especiais

- O CRM deve ser único para cada médico no sistema.
- Em produção os campos de e-mail e telefone são únicos. Nos ambientes de desenvolvimento isso não é aplicado.

- Informações Adicionais

- Esclarecer os formatos esperados para campos específicos, como o formato do CRM, e-mail, telefone, e outros, para garantir a consistência das informações.
- Se aplicável, incluir políticas de senhas para garantir a segurança do acesso ao sistema, especialmente se houver informações sensíveis dos médicos.

- Protótipo Essencial

- Identificar as telas essenciais para o cadastro de médicos, incluindo aquelas que mostram os campos obrigatórios, opcionais e as mensagens de confirmação.
- Descrever o fluxo de interação entre o administrador e o sistema durante o cadastro do médico, destacando pontos cruciais como a verificação de unicidade do CRM.
- Incluir elementos de feedback ao usuário, como mensagens de sucesso, alertas de erro e confirmações, semelhante ao caso de uso anterior.

- [CU-03] Cadastrar paciente

- Descrição Resumida dos Atores

Administrador do sistema – onde o Administrador é um usuário administrador com todas as permissões do sistema

- Pré-condições

Administrador deve estar autenticado

- Cenário Principal

- O Administrador clica no botão “Cadastrar Paciente” no painel administrativo
- O sistema carrega uma página com os seguintes componentes
  - Um formulário contendo os campos:
    - Nome Completo
    - Data de Nascimento
    - Endereço
    - Número de Telefone
    - Medicamentos Contínuos (campo de texto para inserir detalhes)
  - O Administrador preenche os campos obrigatórios com as informações do paciente que deseja cadastrar.
  - O Administrador pode, opcionalmente, preencher campos adicionais, como:
    - E-mail
    - Grupo Sanguíneo
    - Gênero
    - Peso
    - Altura
  - O Administrador verifica se todas as informações estão corretas e completas.
  - Após verificar, o Administrador clica no botão “Cadastrar” e confirmando o cadastro do paciente.

- Cenários Alternativos

- Campos Obrigatórios não preenchidos: Se o Administrador não preencher algum dos campos obrigatórios, o sistema exibe uma mensagem de erro indicando quais campos precisam ser preenchidos.
- Informações Incorretas: Se o Administrador fornecer informações incorretas, o sistema valida os dados e exibe mensagens de erro apropriadas.

- Pós-condições
  - Cadastro Bem-Sucedido: Após a confirmação do cadastro, o sistema armazena as informações do paciente na base de dados.
  - Mensagem de Confirmação: O sistema exibe uma mensagem de confirmação ao Administrador, indicando que o paciente foi cadastrado com sucesso.
- Requisitos Especiais
  - Garantir que o sistema valide se todos os campos obrigatórios foram preenchidos antes de permitir o cadastro do paciente.
- Informações Adicionais
  - Especificar os formatos esperados para dados como Data de Nascimento, Número de Telefone, E-mail, Peso, Altura, etc., para garantir a consistência e integridade dos dados.
  - Incluir mensagens de erro personalizadas para cada cenário alternativo, informando claramente ao administrador o que precisa ser corrigido.
- Protótipo Essencial
  - Identificar as telas essenciais para o cadastro de pacientes, incluindo aquelas que mostram os campos obrigatórios, opcionais e as mensagens de confirmação.
  - Descrever o fluxo de interação entre o administrador e o sistema durante o cadastro do paciente, destacando pontos cruciais como a validação de campos obrigatórios.
  - Incluir elementos de feedback ao usuário, como mensagens de sucesso, alertas de erro e confirmações, semelhante aos casos de uso anteriores.
- [CU-04] Preencher relatório da consulta
  - Descrição Resumida dos Atores
 

Médico – Um profissional de saúde registrado no sistema que realiza consultas com pacientes e registra os detalhes da consulta.
  - Pré-condições
    - O médico deve estar autenticado no sistema.
    - O médico deve ter uma consulta agendada com um paciente.
  - Cenário Principal
    - O Médico clica na consulta correspondente
    - O sistema carrega uma página com os seguintes componentes
      - Um formulário contendo os campos:

- Nome do Paciente (selecionado automaticamente com base na consulta agendada)
  - Sintomas ou Motivo da Consulta (campo de texto)
  - Diagnóstico (campo de texto)
  - Tratamento Recomendado (campo de texto)
  - Prescrição Médica (campo de texto)
  - Observações Adicionais (campo de texto)
  - O médico preenche os campos obrigatórios com as informações da consulta que deseja cadastrar.
  - O médico pode, opcionalmente, preencher campos adicionais, como:
    - Pressão Arterial
    - Peso do Paciente
    - Exames Solicitados (campo de texto para listar exames recomendados)
  - O médico verifica se todas as informações estão corretas e completas.
  - Após verificar, o médico clica no botão “Cadastrar” e confirmando o cadastro no sistema.
- Cenários Alternativos
    - Se o médico encontrar erros ou informações em falta, ele corrige os campos relevantes e tenta novamente registrar a consulta.
  - Pós-condições
    - O resumo da consulta é registrado com sucesso no sistema e vinculado ao paciente e ao médico.
  - Requisitos Especiais
    - O sistema deve permitir o acesso ao histórico de consultas do paciente, incluindo resumos anteriores.
  - Informações Adicionais
    - Especificar os formatos esperados para dados como Pressão Arterial, Peso do Paciente e outros campos relevantes.
    - Se aplicável, detalhar como o sistema lida com a integração de outros sistemas para listar exames recomendados.
  - Protótipo Essencial
    - Identificar as telas essenciais para o preenchimento do relatório da consulta, incluindo aquelas que mostram os campos obrigatórios, opcionais e as mensagens de confirmação.
    - Descrever o fluxo de interação entre o médico e o sistema durante o preenchimento do relatório da consulta, destacando pontos cruciais como a verificação de informações e o registro bem-sucedido.
    - Incluir telas ou elementos de interface que permitam ao médico acessar o histórico de consultas do paciente.
    - Incluir elementos de feedback ao usuário, como mensagens de sucesso, alertas de erro e confirmações.



- [CU-05] Buscar Médicos Credenciados pelo Paciente

- Descrição Resumida dos Atores

Paciente – Um usuário registrado no sistema que procura médicos credenciados com base em critérios específicos.

- Pré-condições

- O paciente deve estar autenticado no sistema.

- Cenário Principal

- O paciente acessa a funcionalidade de pesquisa de médicos em sua aplicação.
- O sistema exibe um formulário de pesquisa que permite ao paciente especificar os seguintes critérios:
  - Cidade (campo de texto ou lista suspensa)
  - Bairro (campo de texto ou lista suspensa)
  - Especialização (campo de texto ou lista suspensa)
- O paciente preenche um ou mais dos critérios de pesquisa mencionados acima.
- O paciente clica no botão “Buscar” ou “Pesquisar”.
- O sistema realiza uma pesquisa com base nos critérios fornecidos pelo paciente.
- O sistema exibe uma lista de médicos credenciados que atendem aos critérios de pesquisa do paciente.
- O paciente pode visualizar informações detalhadas de cada médico, incluindo nome, especialização, endereço, detalhes de contato e horário de atendimento.
- O sistema exibe um botão “Agendar Consulta” ao lado dos dados dos médicos encontrados

- Cenários Alternativos

- Se não houver médicos que atendam aos critérios de pesquisa do paciente, o sistema exibe uma mensagem informando que não foram encontrados resultados.

- Pós-condições

- O resumo da consulta é registrado com sucesso no sistema e vinculado ao paciente e ao médico.

- Requisitos Especiais

- O sistema deve ser capaz de lidar com pesquisas que incluam critérios em branco, o que significa que o paciente pode pesquisar apenas por cidade, apenas por especialização, etc.

- Informações Adicionais

- Especificar os formatos esperados para dados como cidade, bairro, especialização, etc., para garantir a consistência e integridade dos dados.
- Incluir uma mensagem informativa se não houver médicos que atendam aos critérios de pesquisa do paciente.

- Protótipo Essencial

- Identificar as telas essenciais para a busca de médicos, incluindo aquelas que mostram os critérios de pesquisa, resultados e detalhes de médicos.
- Descrever o fluxo de interação entre o paciente e o sistema durante a busca de médicos, destacando pontos cruciais como a visualização de resultados e a opção de agendar consulta.
- Mostrar como o sistema trata pesquisas que incluem critérios em branco e como são apresentados os resultados nesses casos.
- Incluir elementos de feedback ao usuário, como mensagens de sucesso, alertas de erro e informações sobre a falta de resultados.

## • [CU-06] Agendar Consulta pelo Paciente

### • Descrição Resumida dos Atores

Paciente – Um usuário registrado no sistema que deseja agendar uma consulta com um médico credenciado.

- Pré-condições
  - O paciente deve estar autenticado no sistema.
  - O paciente realizou uma busca e selecionou um médico credenciado.
- Cenário Principal
  - Após realizar uma busca e selecionar um médico credenciado, o paciente clica no botão “Agendar Consulta”
  - O sistema exibe um calendário ou formulário de agendamento com os seguintes campos obrigatórios:
    - Data da Consulta (selecionável no calendário)
    - Hora da Consulta (selecionável em uma lista suspensa)
    - Observações ou Sintomas (campo de texto opcional para o paciente inserir informações relevantes)
  - O paciente seleciona uma data e hora disponível para a consulta.
  - O paciente pode inserir observações ou sintomas relevantes (opcional).
  - O paciente verifica se as informações estão corretas e completas.
  - O paciente clica no botão “Confirmar Agendamento” para registrar a consulta.
- Cenários Alternativos
  - Se não houver datas e horários disponíveis para agendamento com o médico selecionado, o sistema informa o paciente e sugere outras datas ou médicos disponíveis.
  - Se o paciente deixar campos obrigatórios em branco ou inserir informações inválidas, o sistema exibe mensagens de erro e solicita que as informações sejam corrigidas.
- Pós-condições
  - A consulta é agendada com sucesso, e o paciente recebe uma confirmação.

- O sistema envia uma notificação ao médico sobre o agendamento.
- O sistema informa que o paciente deve cancelar a consulta com 24h de antecedência. Caso o cliente não o faça, o sistema bloqueia o paciente de agendar a consulta nos próximos 15 dias.
- Requisitos Especiais
  - O sistema deve validar a disponibilidade de datas e horários para o agendamento.
  - Implementar uma política em que o paciente seja informado sobre a necessidade de cancelar a consulta com 24 horas de antecedência.
  - Bloquear o paciente de agendar uma nova consulta nos próximos 15 dias caso não cumpra com a política de cancelamento.
- Informações Adicionais
  - Especificar os formatos esperados para dados como Data da Consulta, Hora da Consulta e quaisquer outros campos relevantes.
  - Incluir mensagens claras e informativas para orientar o paciente durante o processo de agendamento, tanto em caso de sucesso quanto em cenários alternativos.
- Protótipo Essencial
  - Identificar as telas essenciais para o agendamento da consulta, incluindo aquelas que mostram as opções de data e hora, observações e o botão de confirmação.
  - Descrever o fluxo de interação entre o paciente e o sistema durante o agendamento da consulta, destacando pontos cruciais como a validação de disponibilidade e o feedback ao paciente.
  - Incluir informações sobre a política de cancelamento na interface do paciente, alertando sobre as consequências do não cumprimento.
  - Mostrar como o sistema fornece feedback ao paciente após o agendamento bem-sucedido, incluindo informações sobre a notificação ao médico.

## • [CU-07] Visualizar dados da consulta pelo Paciente

### • Descrição Resumida dos Atores

Paciente – Um usuário registrado no sistema que deseja visualizar a listagem de suas consultas e seus estados.

- Pré-condições
  - O paciente deve estar autenticado no sistema.
- Cenário Principal

- O paciente acessa a funcionalidade de listagem de consultas em sua aplicação.
- O sistema exibe uma lista de todas as consultas do paciente, incluindo as seguintes informações:
  - Data e Hora da Consulta
  - Nome do Médico
  - Especialização do Médico
  - Local da Consulta (clínica ou hospital, se aplicável)
  - Estado da Consulta (por exemplo, “Agendada,” “Finalizada,” “Aguardando Exames,” “Exames Prontos,” etc.)
- O paciente pode clicar em uma consulta listada para ver mais detalhes, como resumo da consulta, prescrições, exames solicitados e outras informações relevantes.
- Cenários Alternativos
  - Se o paciente não tiver consultas agendadas ou registradas, o sistema informa o paciente de que não há consultas disponíveis na listagem.
  - Se o paciente deixar campos obrigatórios em branco ou inserir informações inválidas, o sistema exibe mensagens de erro e solicita que as informações sejam corrigidas.
- Pós-condições
- Requisitos Especiais
  - O sistema deve manter registros atualizados sobre o estado de cada consulta, refletindo eventos como agendamento, conclusão de consultas e disponibilidade de resultados de exames.
- Informações Adicionais
  - Especificar os formatos esperados para dados como Data e Hora da Consulta, Nome do Médico, Estado da Consulta, etc., para garantir a consistência e integridade dos dados.
  - Incluir mensagens claras e informativas para orientar o paciente durante a visualização da listagem de consultas, tanto em caso de sucesso quanto em cenários alternativos.
- Protótipo Essencial
  - Identificar as telas essenciais para a visualização da listagem de consultas, incluindo aquelas que mostram as informações básicas e detalhes adicionais de uma consulta específica.
  - Descrever o fluxo de interação entre o paciente e o sistema durante a visualização da listagem de consultas, destacando pontos cruciais como a navegação para detalhes de uma consulta específica.
  - Mostrar como o sistema registra eventos relacionados ao estado de cada consulta e como essas informações são apresentadas ao paciente.
  - Incluir uma mensagem informativa se o paciente não tiver consultas agendadas ou registradas.

- [CU-08] Consultar Agenda dos Médicos

- Descrição Resumida dos Atores

Paciente – Um usuário registrado no sistema que deseja visualizar a agenda do médico que deseja ter a consulta.

- Pré-condições

- O paciente deve estar autenticado no sistema.
- A agenda do(s) médicos(s) deve estar cadastrada no sistema.

- Cenário Principal

- O paciente acessa a funcionalidade visualização da agenda do médico.
- O sistema exibe uma agenda em forma de calendário.
- O paciente clica no dia que deseja consultar e é listados os horários.
- Caso o dia ou o horário esteja indisponível, o campo fica desabilitado para ser selecionado.
- Caso o dia ou o horário esteja disponível, o cliente clica no botão [Agendar Consulta]

- Cenários Alternativos

- Se o médico estiver com todos os horários do dia lotados, o sistema emite uma mensagem “não há horários disponíveis para essa data”.

- Pós-condições

- Se o paciente realizar com sucesso o agendamento de uma consulta para um horário disponível, a consulta é considerada agendada.
- Após o agendamento, o sistema deve atualizar automaticamente a agenda do médico, marcando o horário como ocupado.
- O sistema deve enviar uma notificação ao médico sobre o novo agendamento, incluindo detalhes como nome do paciente, data e hora da consulta.

- Requisitos Especiais

- O sistema deve manter registros atualizados sobre a agenda dos médicos

- Informações Adicionais

- Especificar os formatos esperados para dados como a visualização da agenda, datas, horários disponíveis e indisponíveis.
- Incluir mensagens claras e informativas para orientar o paciente durante a visualização da agenda do médico, especialmente em cenários alternativos, como a falta de horários disponíveis.

- Protótipo Essencial

- Identificar as telas essenciais para a visualização da agenda do médico, incluindo aquelas que mostram o calendário, horários disponíveis e botões de agendamento.
- Descrever o fluxo de interação entre o paciente e o sistema durante a visualização da agenda do médico, destacando pontos cruciais como a seleção de datas e horários.
- Mostrar claramente no protótipo os estados dos horários (disponíveis, indisponíveis) e como o paciente pode interagir com esses horários.

## • [CU-09] Consultar Resultado de Exames

### • Descrição Resumida dos Atores

Paciente – Um usuário registrado no sistema que deseja visualizar a listagem de seus exames.

### • Pré-condições

- O paciente deve estar autenticado no sistema.

### • Cenário Principal

1. O paciente acessa a funcionalidade de consulta de exames.
2. O sistema exibe uma lista de todos os exames realizados pelos pacientes, incluindo as seguintes informações:
  1. Data e Hora da Exame
  2. Nome do Médico Solicitante
  3. Local do Exame (clínica ou hospital, se aplicável)
3. O paciente pode clicar em um exame listado para ver mais detalhes.

### • Cenários Alternativos

- Se o paciente não tiver exames registrados, o sistema informa o paciente de que não há resultados disponíveis.

### • Pós-condições

- Após consultar com sucesso os resultados de um exame, o paciente pode visualizar informações detalhadas, como data e hora do exame, nome do médico solicitante e resultados específicos.

### • Requisitos Especiais

- O sistema deve manter registros atualizados sobre os resultados de exames.

### • Informações Adicionais

- Especificar os formatos esperados para dados como Data e Hora do Exame, Nome do Médico Solicitante, Local do Exame, etc., para garantir a consistência e integridade dos dados.

- Incluir mensagens claras e informativas para orientar o paciente durante a visualização da listagem de exames, tanto em caso de sucesso quanto em cenários alternativos, como a falta de resultados disponíveis.
- Protótipo Essencial
  - Identificar as telas essenciais para a visualização da listagem de exames, incluindo aquelas que mostram as informações básicas e detalhes adicionais de um exame específico.
  - Descrever o fluxo de interação entre o paciente e o sistema durante a visualização da listagem de exames, destacando pontos cruciais como a seleção de exames específicos para visualização detalhada.
  - Incluir uma mensagem informativa se o paciente não tiver exames registrados ou se os resultados ainda não estiverem disponíveis.
- [CU-10] Bloquear Consultas
  - Descrição Resumida dos Atores
 

Médico – Um usuário registrado no sistema que deseja bloquear consultas.
  - Pré-condições
    - O médico deve estar autenticado no sistema.
  - Cenário Principal
    1. O médico acessa a funcionalidade de agenda de consultas.
    2. O sistema exibe calendário do mês exibindo os dias e horários.
    3. O médico seleciona o(s) dia(s) ou o(s) horário(s) em que deseja bloquear.
  - Cenários Alternativos
    - **Cancelamento de Bloqueio:** Se o médico desejar desbloquear um dia ou horário previamente bloqueado:
      - O médico acessa a funcionalidade de agenda de consultas.
      - O sistema exibe um calendário do mês, destacando os dias ou horários já bloqueados.
      - O médico seleciona a opção de desbloquear para o(s) dia(s) ou horário(s) desejado(s).
      - O sistema confirma o desbloqueio e atualiza a agenda do médico.
    - **Conflito com Consultas Agendadas:** Se o médico tentar bloquear um dia ou horário que já possui consultas agendadas:
      - O sistema exibe uma mensagem de alerta informando que há consultas agendadas para o(s) dia(s) ou horário(s) selecionado(s).

- O médico pode optar por desmarcar as consultas agendadas antes de prosseguir com o bloqueio, ou cancelar a ação de bloqueio.
- Pós-condições
  - **Bloqueio Atualizado:** Após a ação de bloquear consultas, o sistema mantém a agenda do médico atualizada, refletindo os dias ou horários bloqueados.
  - **Desbloqueio Atualizado:** Se o médico desbloquear consultas, o sistema mantém a agenda do médico atualizada, removendo os dias ou horários desbloqueados.
- Requisitos Especiais
  - O sistema deve garantir que a agenda do médico seja consistente, refletindo corretamente os dias ou horários bloqueados.
- Informações Adicionais
  - Incluir mensagens claras e informativas para orientar o médico durante o processo de bloqueio e desbloqueio de consultas, especialmente em cenários alternativos, como tentativas de bloqueio em horários com consultas agendadas.
  - No protótipo, destacar visualmente os dias ou horários bloqueados na interface da agenda do médico para facilitar a identificação.
- Protótipo Essencial
  - Identificar as telas essenciais para o médico bloquear e desbloquear consultas, incluindo aquelas que mostram o calendário do mês e as opções de bloqueio.
  - Mostrar claramente no protótipo mensagens de alerta em casos de tentativa de bloqueio em horários com consultas agendadas, fornecendo opções claras para resolver conflitos.

## • [CU-11] Cancelar Consulta

### • Descrição Resumida dos Atores

Paciente – Um usuário registrado no sistema que deseja cancelar uma consulta agendada.

- Pré-condições
  - O paciente deve estar autenticado no sistema.
- Cenário Principal
  1. O paciente acessa a funcionalidade de consulta de exames.
  2. O sistema dos exames agendados
  3. O paciente pode clicar em um exame listado para cancelar.



- Cenários Alternativos
  - Se o paciente não tiver exames agendados, o sistema informa o paciente de que não há exames disponíveis para cancelar.
  - O sistema emite uma mensagem informando se cliente realmente deseja cancelar a consulta
  - Caso o cliente esteja cancelando a consulta com menos de 24h para a mesma, o sistema informa que se cliente prosseguir com cancelamento só conseguirá agendar novamente após 15 dias.

- Pós-condições

**Consulta Cancelada:** Após a confirmação do cancelamento, o sistema atualiza os registros, cancelando a consulta agendada.

**Restrição de Agendamento Aplicada:** Se o paciente cancelou a consulta com menos de 24 horas para a mesma, o sistema aplica a restrição de 15 dias para o próximo agendamento.

- Requisitos Especiais
  - O sistema deve manter registros atualizados sobre os resultados de exames.
- Informações Adicionais
 

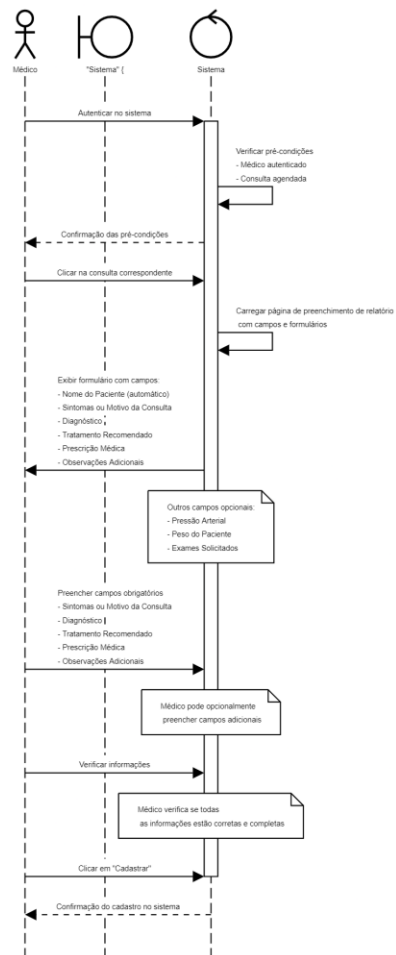
O sistema pode enviar um e-mail de confirmação para o paciente após o cancelamento, contendo informações relevantes.

- Protótipo Essencial

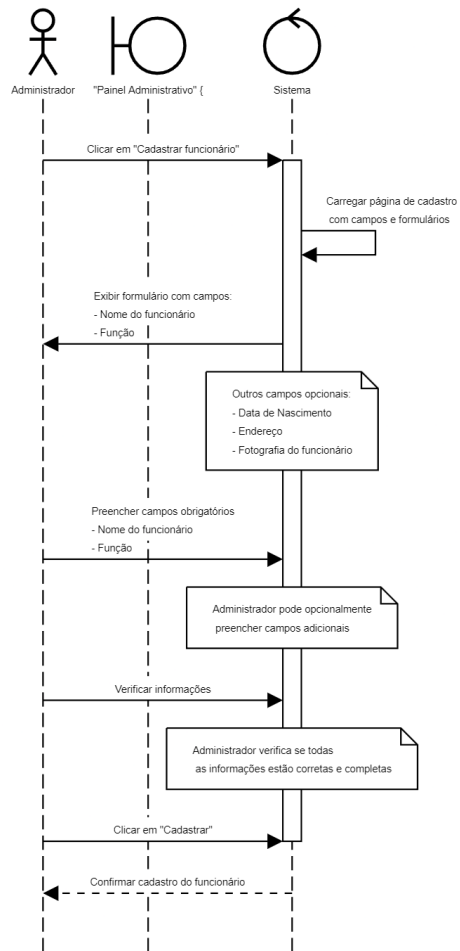
- Identificar as telas essenciais para o paciente visualizar e cancelar consultas agendadas, incluindo aquelas que mostram a listagem de consultas e as opções de cancelamento.
- Descrever o fluxo de interação entre o paciente e o sistema durante o processo de cancelamento, destacando pontos cruciais, como a confirmação da ação e a aplicação de restrições.
- Mostrar claramente no protótipo mensagens de alerta caso o paciente tente cancelar uma consulta com menos de 24 horas de antecedência, explicando a restrição de agendamento subsequente.

## Diagrama de Sequência

Autenticar no Sistema:

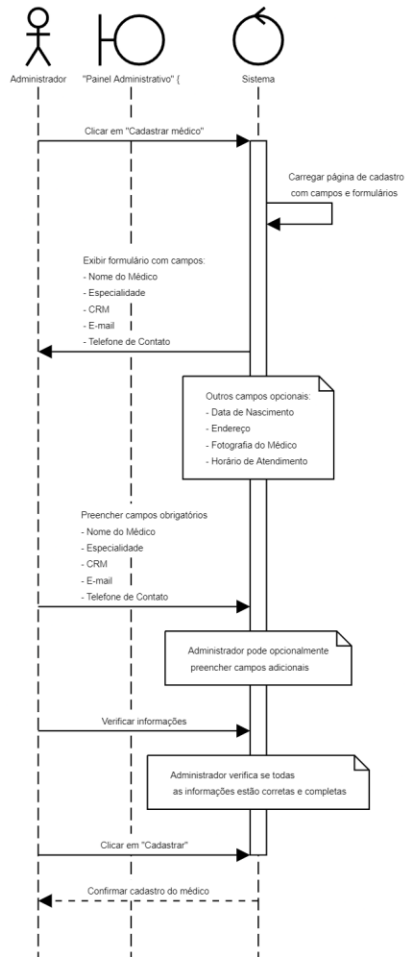


Cadastrar Funcionário:

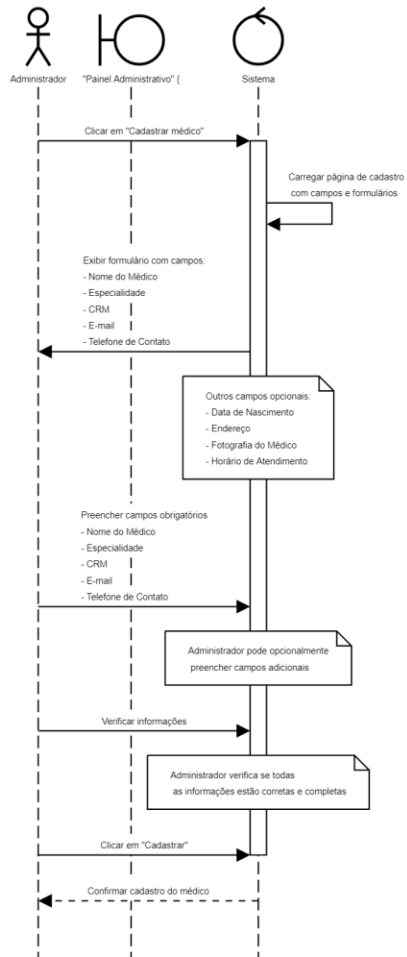


]

Cadastro Médico:

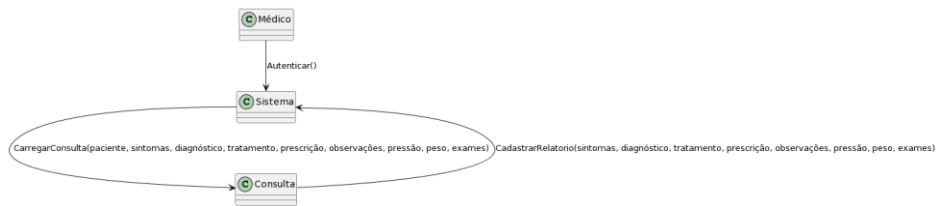


Cadastro Paciente:

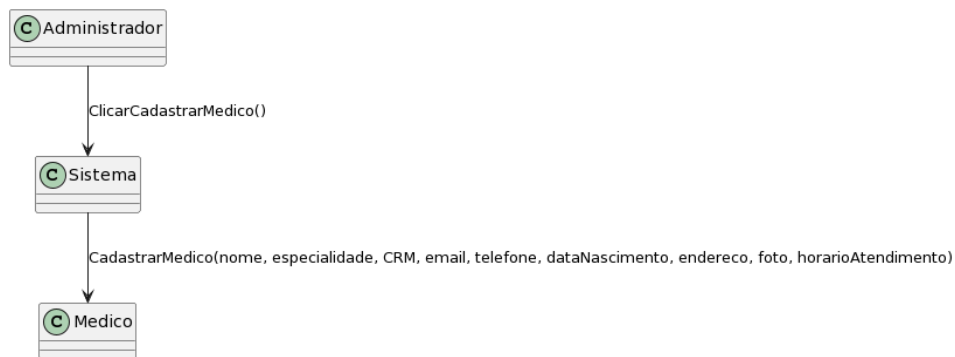


## Diagrama de Classes

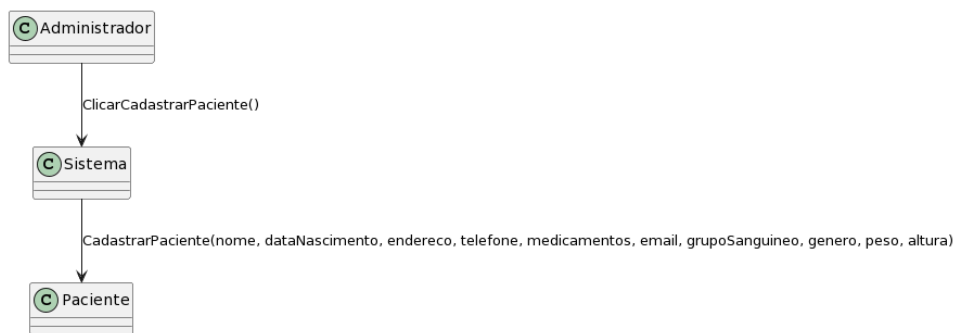
### Autenticar no Sistema:



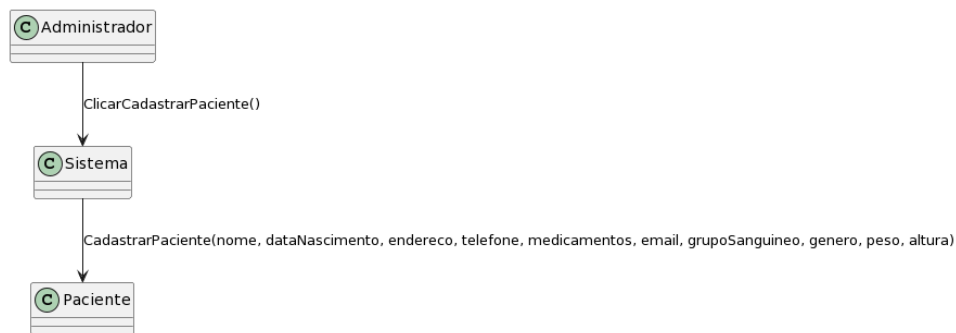
### Cadastrar Médico:



### Cadastrar Paciente:



Carregar Consulta:



## Mapeamento do Diagrama de Classes para Código em Java

Autenticar no Sistema:

```
public class Médico {
    // Método que simula a autenticação do médico no sistema
    public void Autenticar() {
        System.out.println("Médico autenticado no sistema.");
        // Implementação fictícia da autenticação do médico
    }
}
```

```
public class Sistema {
    // Método que simula o carregamento da consulta e o cadastro do relatório
    public void CarregarConsulta(Paciente paciente, String sintomas, String diagnostico, String tratamento,
                                String prescricao, String observacoes, String pressao, String peso, String exames) {
        System.out.println("Carregando consulta para o paciente: " + paciente.getNome());
        System.out.println("Sintomas: " + sintomas);
        System.out.println("Diagnóstico: " + diagnostico);
        System.out.println("Tratamento: " + tratamento);
        System.out.println("Prescrição Médica: " + prescricao);
        System.out.println("Observações: " + observacoes);
        System.out.println("Pressão Arterial: " + pressao);
    }
}
```

```

        System.out.println("Peso: " + peso);
        System.out.println("Exames Solicitados: " + exames);
        // Implementação fictícia do carregamento da consulta e do cadastro do relatório
    }
}

public class Consulta {
    // Atributos fictícios da classe Consulta

    // Método fictício que simula o cadastro do relatório da consulta
    public void CadastrarRelatorio(String sintomas, String diagnostico, String tratamento, String
prescricao,
                                String observacoes, String pressao, String peso, String exames) {
        System.out.println("Cadastro do relatório da consulta:");
        System.out.println("Sintomas: " + sintomas);
        System.out.println("Diagnóstico: " + diagnostico);
        System.out.println("Tratamento: " + tratamento);
        System.out.println("Prescrição Médica: " + prescricao);
        System.out.println("Observações: " + observacoes);
        System.out.println("Pressão Arterial: " + pressao);
        System.out.println("Peso: " + peso);
        System.out.println("Exames Solicitados: " + exames);
        // Implementação fictícia do cadastro do relatório da consulta
    }
}

```

Cadastrar Médico:

```

public class Administrador {
    // Método que simula o clique em "Cadastrar Médico"
    public void ClicarCadastrarMedico() {

```



```

        System.out.println("Botão 'Cadastrar Médico' clicado.");
        // Implementação fictícia do clique em "Cadastrar Médico"
    }
}

public class Sistema {
    // Método que simula o cadastro de um médico

    public void CadastrarMedico(String nome, String especialidade, String CRM, String email,
String telefone,
                                String dataNascimento, String endereco, String foto, String
horarioAtendimento) {
        System.out.println("Cadastrando médico:");
        System.out.println("Nome: " + nome);
        System.out.println("Especialidade: " + especialidade);
        System.out.println("CRM: " + CRM);
        System.out.println("Email: " + email);
        System.out.println("Telefone: " + telefone);
        System.out.println("Data de Nascimento: " + dataNascimento);
        System.out.println("Endereço: " + endereco);
        System.out.println("Foto: " + foto);
        System.out.println("Horário de Atendimento: " + horarioAtendimento);
        // Implementação fictícia do cadastro do médico
    }
}

public class Medico {
    // Atributos fictícios da classe Medico

    // Métodos fictícios da classe Medico
}

Cadastrar Paciente:
public class Administrador {
    // Método que simula o clique em "Cadastrar Paciente"

    public void ClicarCadastrarPaciente() {

```

```

        System.out.println("Botão 'Cadastrar Paciente' clicado.");
        // Implementação fictícia do clique em "Cadastrar Paciente"
    }
}

public class Sistema {
    // Método que simula o cadastro de um paciente
    public void CadastrarPaciente(String nome, String dataNascimento, String endereco, String
telefone,
                                String medicamentos, String email, String grupoSanguineo, String genero,
                                String peso, String altura) {
        System.out.println("Cadastrando paciente:");
        System.out.println("Nome Completo: " + nome);
        System.out.println("Data de Nascimento: " + dataNascimento);
        System.out.println("Endereço: " + endereco);
        System.out.println("Número de Telefone: " + telefone);
        System.out.println("Medicamentos Contínuos: " + medicamentos);
        System.out.println("Email: " + email);
        System.out.println("Grupo Sanguíneo: " + grupoSanguineo);
        System.out.println("Gênero: " + genero);
        System.out.println("Peso: " + peso);
        System.out.println("Altura: " + altura);
        // Implementação fictícia do cadastro do paciente
    }
}

public class Paciente {
    // Atributos fictícios da classe Paciente

    // Métodos fictícios da classe Paciente
}

```

Carregar Consulta:

```

public class Médico {
    // Método que simula a autenticação do médico no sistema
    public void Autenticar() {
        System.out.println("Médico autenticado no sistema.");
        // Implementação fictícia da autenticação do médico
    }
}

public class Sistema {
    // Método que simula o carregamento da consulta e o cadastro do relatório
    public void CarregarConsulta(Paciente paciente, String sintomas, String diagnostico, String
tratamento,
                                String prescricao, String observacoes, String pressao, String peso, String
exames) {
        System.out.println("Carregando consulta para o paciente: " + paciente.getNome());
        System.out.println("Sintomas: " + sintomas);
        System.out.println("Diagnóstico: " + diagnostico);
        System.out.println("Tratamento: " + tratamento);
        System.out.println("Prescrição Médica: " + prescricao);
        System.out.println("Observações: " + observacoes);
        System.out.println("Pressão Arterial: " + pressao);
        System.out.println("Peso: " + peso);
        System.out.println("Exames Solicitados: " + exames);
        // Implementação fictícia do carregamento da consulta e do cadastro do relatório
    }
}

public class Consulta {
    // Atributos fictícios da classe Consulta

    // Método fictício que simula o cadastro do relatório da consulta
    public void CadastrarRelatorio(String sintomas, String diagnostico, String tratamento, String
prescricao,

```

```

        String observacoes, String pressao, String peso, String exames) {
    System.out.println("Cadastro do relatório da consulta:");
    System.out.println("Sintomas: " + sintomas);
    System.out.println("Diagnóstico: " + diagnostico);
    System.out.println("Tratamento: " + tratamento);
    System.out.println("Prescrição Médica: " + prescricao);
    System.out.println("Observações: " + observacoes);
    System.out.println("Pressão Arterial: " + pressao);
    System.out.println("Peso: " + peso);
    System.out.println("Exames Solicitados: " + exames);
    // Implementação fictícia do cadastro do relatório da consulta
}

}

public class Paciente {
    // Atributos fictícios da classe Paciente

    // Métodos fictícios da classe Paciente
}

```