

<Gestão de exames Laboratoriais>

Documento de Requisitos
Versão 2.0

Responsável: Daniel Lucas Soares Souza Farias
E-mail: daniellssf@hotmail.com

Histórico de Revisões

Data	Versão	Descrição	Autor
01/04/2021	1.0	Versão inicial do documento	Daniel Lucas Farias
08/06/2021	2.0	Versão final do documento	Daniel Lucas Farias

Gestão de exames Laboratoriais

Documento de Requisitos

1 Introdução

1.1 Propósito do documento de requisitos

Este documento destina-se aos clientes, engenheiros e gerentes envolvidos no desenvolvimento do sistema, doravante referido apenas como Gestão de exames Laboratoriais. O propósito deste documento é apresentar a descrição dos serviços e funções que o sistema a ser desenvolvido deve prover, bem como as suas restrições de operação e propriedades gerais, a fim de ilustrar uma descrição detalhada do sistema para um auxílio durante as etapas de análise, projeto e testes. O documento especifica todos os requisitos funcionais e não funcionais do sistema e foi preparado levando-se em conta as funcionalidades levantadas durante a fase de concepção do sistema.

1.2 Escopo do produto

O projeto consiste na construção de um sistema (aplicação web) cujo objetivo é gerenciar os exames laboratoriais dos clientes de uma determinada clínica. Este sistema visa auxiliar a consulta de datas de entrega dos exames, controle sobre os clientes e exames.

1.3 Concepção do sistema

Foram usados dois métodos para que pudessem ser obtidos os requisitos do sistema:

- Consulta com especialista:
 - Prof. Thiago Baesso Procaci, engenheiro de software e docente no Instituto de Tecnologia INFNET, orientou a análise de requisitos devido a sua grande experiência em desenvolvimento de software;
- Entrevista com donos de clínicas veterinárias que oferecem serviços de exame de laboratório.

1.4 Convenções, Termos e Abreviações

Para evitar interpretações incorretas deste documento, algumas convenções e termos específicos são descritos a seguir:

JDK	Java Development Kit
-----	----------------------

1.4.1 Identificação dos Requisitos

Cada requisito será unicamente identificado no formato [tipoRequisito.numero]. Para requisitos funcionais, o código do tipo de requisito será RF, e para requisitos não funcionais, RNF. Um número será assinalado a cada requisito de forma incremental, na ordem que forem mencionados neste documento.

1.4.2 Prioridade dos Requisitos

Foram adotadas as seguintes denominações para estabelecer a prioridade dos requisitos: essencial, importante e desejável.

- **Essencial:** é o requisito sem o qual o sistema não entra em funcionamento, ou seja, são requisitos imprescindíveis tendo que ser implementados impreterivelmente.
- **Importante:** é o requisito sem o qual o sistema entra em funcionamento, mas de maneira insatisfatória, ou seja, devem ser implementados, mas se não forem, o sistema poderá ser implantado e usado mesmo assim.
- **Desejável:** é o requisito que não compromete as funcionalidades básicas do sistema, podendo funcionar de forma satisfatória sem ele, ou seja, são requisitos que podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

1.5 Referências

Esta subseção apresenta as referências aos documentos que utilizamos no auxílio à construção deste documento de requisitos.

- Periódicos da CAPES - <http://www.periodicos.capes.gov.br/>
- Engenharia de Software - Uma Abordagem Profissional - 8ª Ed. 2016 - Pressman, Roger S.
- Página da Disciplina Especificação de Requisitos e Validação de Sistemas-
<http://www.cin.ufpe.br/~if716/>
- Material do Moodle da Disciplina de Orientação a Objetos JAVA do instituto INFNET
- Material do Moodle da Disciplina de Projeto de Bloco JAVA do instituto INFNET

1.6 Visão Geral

Este documento está organizado da seguinte forma:

- A seção 1 apresentou uma introdução ao documento de requisitos e ao sistema sendo especificado;
- A seção 2 apresenta uma descrição geral do sistema;
- A seção 3 apresenta as definições dos requisitos funcionais e não-funcionais do sistema;
- A seção 4 apresenta o diagrama de casos de uso do sistema, bem como as descrições dos casos de uso definidos;

2 Descrição geral

2.1 Usuários do sistema

O sistema de gestão de exames laboratoriais terá somente dois tipos de usuário: Administrador e Clientes. Ambos terão acesso ao sistema via Login.

2.2 Abrangência e sistemas similares

Abrangência:

O sistema irá conter ferramentas para construção de uma aplicação web que esteja de acordo com os objetivos e metodologia de um sistema de gestão de exames laboratoriais, cujo objetivo consiste em gerenciar os clientes e seus exames. O usuário através de ferramentas como (gráficos estatísticos, documentos, relatórios etc.) poderá realizar atividades que favoreçam a produtividade e análise de prazos, além de gerenciar os exames e clientes, bem como imprimir relatórios. Todo controle de acesso ao sistema será feito por meio de login/senha.

Sistemas similares:

Há diversos sistemas similares que usamos como referência, são eles:

ERPNext (módulo de gerenciamento de laboratório) - O ERPNext é um software Enterprise Resource que oferece aos usuários o controle completo de seus negócios e possui vários módulos. O Módulo de Gerenciamento de Laboratório ERPNext é o que nos interessa.

O ERPNext Lab Management Module é construído em Python, MySQL e NodeJS e possui uma GUI responsiva. Também possui módulos integrados para gerenciamento de estoque, faturamento de pacientes, consultas, consultas, etc.

Bika LIMS - O Bika LIMS é de longe o "OS LIMS com o melhor suporte profissional" e, dada a sua grande base de usuários, é difícil duvidar de sua confiabilidade. É o LIMS de código aberto mais antigo e mais popular com vários derivados, como o Bika Water, o Bika Cannabis LIMS, o Bika Health e o Bika Interlab. O Bika Health é baseado na web, com uma interface de usuário responsiva para uso em dispositivos móveis e tablets e um foco nas ferramentas necessárias para inventário de laboratórios, catalogação, relatórios etc.

Open LIMS - Open LIMS foi registrado em 2010 e não foi atualizado desde 2016, mas como sua contagem de downloads continua aumentando, os desenvolvedores devem ter feito algumas coisas certas.

Ele foi desenvolvido usando PHP e PostgreSQL para rodar em servidores Linux e em qualquer navegador web moderno e possui uma GUI amigável. Você pode estender seus recursos usando extensões gratuitas para diferentes tipos de projetos.

2.3 Suposições e dependências

As seguintes suposições são válidas no decorrer do desenvolvimento do sistema sendo especificado:

- O cliente está responsável pela aquisição de infraestrutura necessária em seu ambiente de produção;
- O cliente será responsável pela disponibilização de recursos de hardware, software, e outros requerimentos destinados à implantação do sistema desenvolvido.

3 Requisitos do Software

3.1 Requisitos Funcionais

<RF01><Efetuar Login>

O sistema deve permitir que o usuário possa efetuar o login com suas credenciais (usuário e senha).

<RF02>< Cadastrar Cliente>

O sistema deve permitir a inserção de um novo Cliente, informando um username do cliente, que deve ser único, username que será usado para efetuar o login, senha usada para efetuar o login, exame realizado e o dia da entrega dos exames.

<RF03>< Remover Cliente>

O sistema deve permitir que clientes e exames sejam deletados, isto é, removidas do cadastro.

<RF04>< Alterar Cliente>

O sistema deve permitir alterar dados do cliente.

<RF05>< Listar Cliente>

Deve ser possível cadastrar novos clientes no sistema.

3.2 Requisitos Não-funcionais

Os requisitos que descrevem os aspectos não-funcionais do sistema são apresentados a seguir:

3.2.1 Requisitos de Segurança

Ident.	Descrição
RNF/SEG-01	O usuário autorizado deverá efetuar <i>login</i> no sistema para poder realizar todas as operações.

3.2.2 Requisitos de Interface

Ident.	Descrição
RNF/INT-01	O sistema deve ter uma interface amigável, de fácil utilização, ao seja, autoexplicável.

3.2.3 Requisitos de Confiabilidade

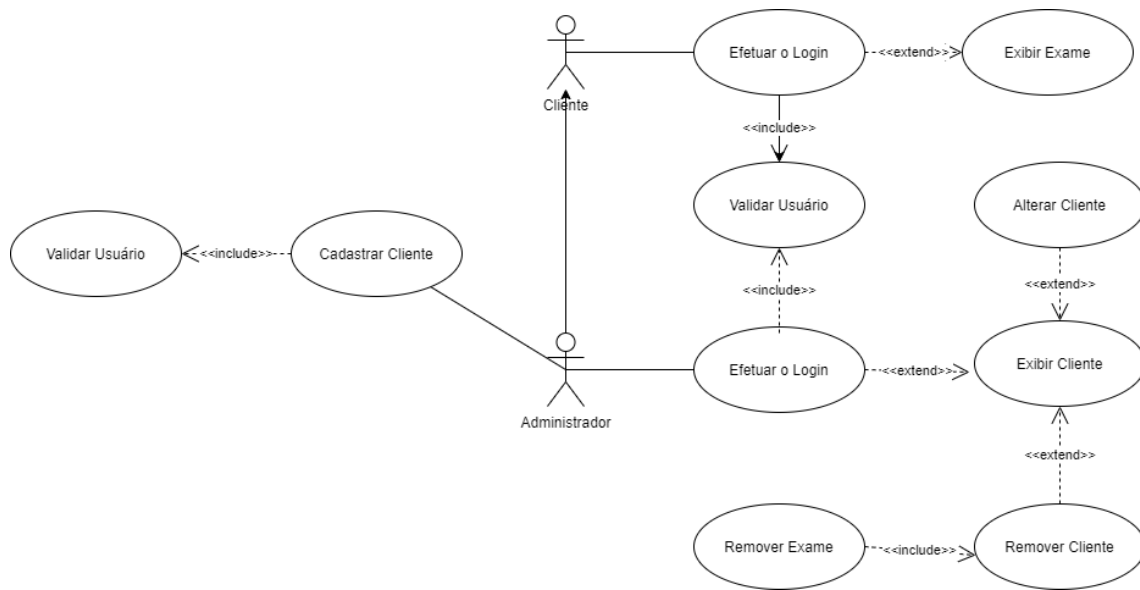
Ident.	Descrição
RNF/CON-01	O sistema deve estar disponível 24 horas por dia durante os 7 dias da semana. Por não se tratar de um sistema crítico, poderá ficar fora do ar até que seja corrigida alguma falha que possa ocorrer.

4 Casos de uso

4.1 Diagrama de casos de uso

O diagrama de casos de uso, expresso em UML (*Unified Modeling Language*), expressa os requisitos funcionais do sistema na forma de casos de uso. Segundo o RUP (*Rational Unified Process*), para cada requisito funcional tem-se um caso de uso. A descrição textual detalhada dos requisitos funcionais, seus fluxos de atividades e requisitos não funcionais associados pode ser encontrada na próxima seção. Na figura abaixo mostramos a representação gráfica em UML dos casos de uso do sistema.

Diagrama de Casos de Uso:



4.2 Descrição dos casos de uso

Dos casos de uso do sistema mostrados no diagrama de casos de uso, foram escolhidos 5 para serem detalhados e trabalhados nas fases de análise e projeto do sistema.

4.2.1 Validar Usuário

[CDU-01]		
Nome:	Validar Usuário	
Atores:	Administrador e cliente	
Prioridade:	Essencial	
Entradas e pré-condições:		<ul style="list-style-type: none">• O usuário não deve estar logado no sistema.
Saídas e pós-condições:		<ul style="list-style-type: none">• O sistema informa se os dados informados estão preenchidos corretamente.
Fluxos de eventos		
Fluxo principal:	<ol style="list-style-type: none">1. O nome e a senha são informados ao sistema.2. Se o nome e senha estiverem de acordo com o registrado no banco de dados do sistema, então o acesso ao sistema é concedido.	
Fluxo Alternativo:	<ol style="list-style-type: none">1. O nome e a senha são informados ao sistema.2. Se o nome e senha não estiverem de acordo com o registrado no banco de dados do sistema, o sistema enviará uma mensagem informando que o usuário ou a senha são inválidos.	

4.2.2 Cadastrar Cliente

[CDU-02]		
Nome:	Cadastrar Cliente	
Atores:	Administrador	
Prioridade:	Essencial	
Entradas e pré-condições:		<ul style="list-style-type: none">• O usuário deve ser o Administrador e deve estar logado no sistema.
Saídas e pós-condições:		<ul style="list-style-type: none">• O sistema informa se os dados informados estão preenchidos corretamente.
Fluxos de eventos		
Fluxo principal:	<ol style="list-style-type: none">1. O sistema exibe uma tela com os campos nome do cliente, login do cliente, senha do cliente, exame do cliente e data de realização do exame.2. Quando acionado o botão cadastrar, o caso de uso Validar Cliente é invocado para realizar a validação.3. Se o cliente for validado com sucesso, insere o cliente na base de dados do sistema e exibe uma mensagem de sucesso para o usuário. Um cliente cadastrado deve ter um ID atribuído automaticamente pelo sistema.	

4.2.3 Alterar Cliente

[CDU-03]		
Nome:	Alterar Cliente	
Atores:	Administrador	
Prioridade:	Essencial	
Entradas e pré-condições:		<ul style="list-style-type: none">• O cliente deve estar cadastrado no sistema.
Saídas e pós-condições:		<ul style="list-style-type: none">• O cliente é atualizado na base de dados com as novas informações.
Fluxos de eventos		
Fluxo principal:	<ol style="list-style-type: none">1. O sistema exibe uma tela com os campos nome do cliente, login do cliente, senha do cliente, exame do cliente e data de realização do exame.2. Quando acionado o botão salvar, o caso de uso Validar Cliente é invocado para realizar a validação.3. Se o cliente for validado com sucesso, insere o cliente na base de dados do sistema e exibe uma mensagem de sucesso para o usuário. Um cliente cadastrado deve ter um ID atribuído automaticamente pelo sistema.	

4.2.4 Remover Cliente

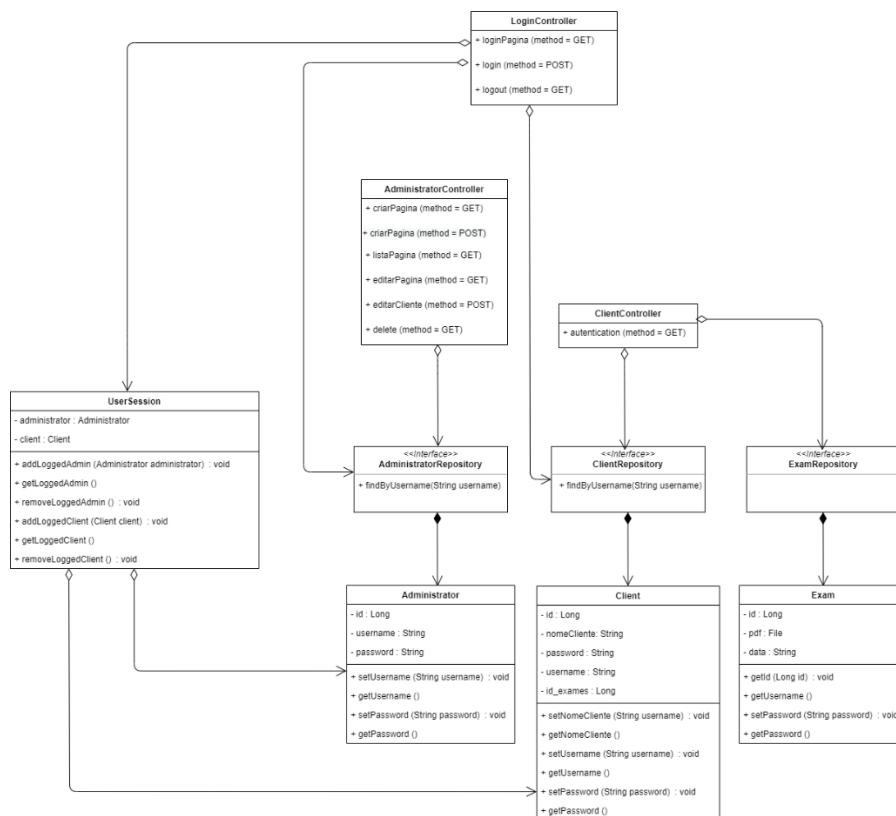
[CDU-04]		
Nome:	Remover Cliente	
Atores:	Administrador	
Prioridade:	Essencial	
Entradas e pré-condições:		<ul style="list-style-type: none">• O Cliente deve estar cadastrado no sistema.
Saídas e pós-condições:		<ul style="list-style-type: none">• O Cliente é removido da base de dados com as novas informações.
Fluxos de eventos		
Fluxo principal:	<ol style="list-style-type: none">1. É informado para o sistema o nome do cliente, login do cliente, senha do cliente, exame do cliente e data de realização do exame.2. O sistema remove o cliente e exibe uma mensagem de sucesso.	

4.2.5 Exibir Cliente

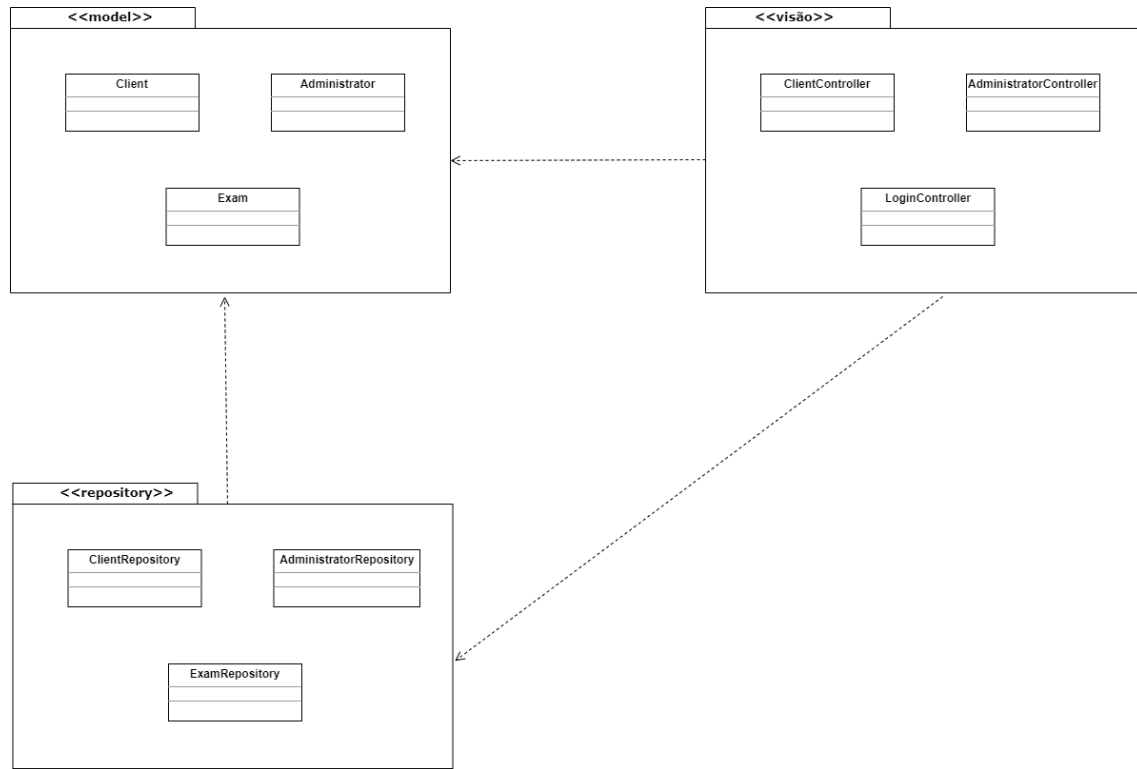
[CDU-05]

Nome:	Exibir Cliente
Atores:	Administrador
Prioridade:	Essencial
Entradas e pré-condições:	<ul style="list-style-type: none"> O Cliente deve estar cadastrado no sistema.
Saídas e pós-condições:	<ul style="list-style-type: none"> O Cliente é removido ou atualizado da base de dados sistema.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> O sistema exibe o ID, nome do cliente, login do cliente, senha do cliente, exame do cliente e data de realização do exame. O Id não poderá ser modificado. O usuário clica no botão Editar e o caso de uso Alterar Cliente é acionado. É informado para o caso de uso acionado o ID, nome do cliente, login do cliente, senha do cliente, exame do cliente e data de realização do exame.
Fluxo Alternativo:	<ol style="list-style-type: none"> O sistema exibe o ID, nome do cliente, login do cliente, senha do cliente, exame do cliente e data de realização do exame. O Id não poderá ser modificado. O usuário clica no botão Deletar e o caso de uso Remover Cliente é acionado. É informado para o caso de uso acionado o ID, nome do cliente, login do cliente, senha do cliente, exame do cliente e data de realização do exame.

5 Diagrama de Classes

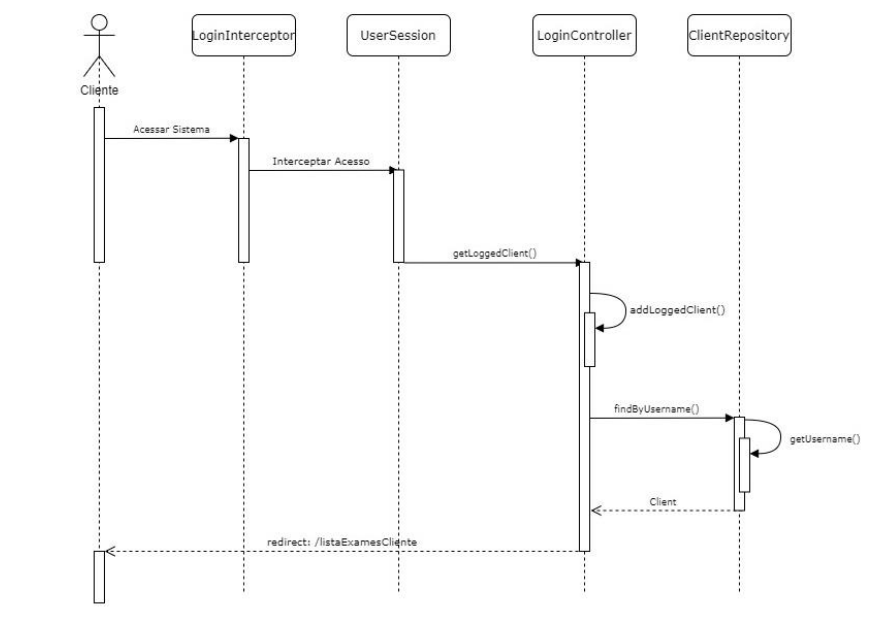


6 Diagrama de Pacotes



7 Diagrama de Sequência

Caso de Uso (Efetuar Login)



Bibliografia:

- <https://lms.infnet.edu.br/moodle/mod/page/view.php?id=226636>
- <https://lms.infnet.edu.br/moodle/mod/page/view.php?id=226644>