Documento de Requisitos de Sistema

**Instituição:** Universidade de São Paulo

**Nome do projeto:** Sistema de Leilão

**Autores:**

* Alberto Lages
* Thaisa Barros
* Uéviton Assunção Santos
* Patrick Douglas
* Gabriel Felipe da Silva
* Anderson Barbosa Cunha
* Gabriel de Moraes Macedo
* Caique Evaristo Santos da Silva
* Leandro de Oliveira Santos

**Orientador:** Rodolfo Riyoei Goya

**Local e Data:** São Paulo,1de novembro 2022

# Prefácio

Essa é a primeira versão do sistema de leilão. Essa solução foi desenvolvida tanto para pessoas físicas quanto jurídicas com o intuito de obter o melhor valor possível na venda dos produtos ofertados no sistema. Nesse software é possível cadastrar um produto para venda ou comprar um produto já cadastrado de acordo com a necessidade do cliente.

# Introdução

O objetivo deste documento é esclarecer aos clientes o funcionamento do Sistema de Leilão. O sistema foi criado para atender as necessidades de venda remunerada de maneira simples, prática e rápida para uma pessoa ou instituição, permitindo o cadastro de produtos a serem leiloados de forma segura e teremos uma restrição para divulgação de produtos lícitos. Mas como funciona? O cliente irá criar uma conta de acesso através de um pequeno formulário disponível na tela principal, o processo de cadastro é simples, será solicitado a validação de alguns dados como CPF, Nome, e-mail, data de nascimento, telefone e endereço e após criar seu acesso ele autenticará no sistema e após isso ele irá decidir se irá fazer uma oferta ou cadastrar um produto para o leilão. Caso o cliente queira cadastrar uma oferta ele será direcionado para uma tela onde será solicitado o nome do produto a ser ofertado, a descrição e o valor mínimo; Caso o cliente queira realizar um lance de um produto ele será direcionado direto para a tela de produtos disponíveis, após isso ele poderá escolher o produto e fazer a oferta que desejar.

# Glossário

Define os termos técnicos usados no documento. Não se deve fazer suposições sobre a experiência ou o conhecimento do leitor.

# Definição de Requisitos de Usuário

- O sistema irá permitir o cadastro de usuários com nome, e-mail e senha.

- O sistema irá permitir o login dos usuários através do e-mail e senha.

- O sistema irá permitir que um usuário cadastre um produto para ser leiloado.

- O sistema não apresentará a outros usuários dados de cunho privativo.

- O sistema deverá guardar os dados de usuários de forma segura.

- O sistema não deverá exceder o tempo de resposta de 10s pra cada requisição.

- O sistema deverá não permitir que um produto cadastrado por um usuário seja leiloado para o mesmo.

- O sistema deverá ter alta disponibilidade, por exemplo, 99% do tempo.

- O sistema não deverá permitir que um lance de um usuário em um produto seja menor que o valor especificado pelo leiloeiro.

- O sistema irá fornecer suporte ao usuário após êxito na compra de um produto leiloado.

- O sistema não irá disponibilizar meios de pagamentos na plataforma.

- O sistema não irá disponibilizar meios de entrega para produtos adquiridos na plataforma.

- O sistema terá tempo limite para cada leilão.

DIAGRAMAS DO SISTEMA

Diagrama

Descrição gerada automaticamente

Diagrama

Descrição gerada automaticamente com confiança média

Uma imagem contendo Diagrama

Descrição gerada automaticamente

Gráfico, Gráfico de cascata

Descrição gerada automaticamente

# Arquitetura do Sistema

O sistema será desenvolvido na arquitetura Cliente/Servidor padrão MVC (Model, View, Controller) , como uma aplicação Web Services API REST (Representational State Transfer) utilizando a linguagem JAVA e o framework Spring Boot, formada por clientes, servidores e recursos, com solicitações gerenciadas por HTTP. O funcionamento será através da Aplicação Cliente (Front-end): páginas estáticas e dinâmicas, Aplicação Servidor(Back-end): entidades, controllers, services, repositories, e base de dados.

Aplicação Cliente: é o componente solicitante de um serviço e envia solicitações para vários tipos de serviços ao servidor. As solicitações feitas pelo cliente ao servidor serão realizadas através de métodos HTTP, como: GET, POST, DELETE, PUT entre outros.

Aplicação Servidor: É o componente que é o provedor de serviços e fornece continuamente serviços ao cliente conforme as solicitações. As solicitações HTTP são recebidas e processada s pelo servidor e ele pode executar retornar alguma tarefa solicitada pelo cliente.

Entity - Entity representa os objetos do banco de dados mapeados em classes, ou seja, funcionam como espécies de espelhos da base. Essa camada, possui apenas os atributos e seus devidos recursos de getters e setters com as referências para as colunas das consultas feitas na base de dados.

Controller - O primeiro componente é o controller, ele é responsável por fazer a interface como usuário, ou seja, é ele quem recebe os parâmetros de entrada, faz as devidas validações de dados, repassa para os próximas componentes e aguarda o processamento para fazer o retorno ao usuário.

Service - O componente service é responsável por encapsular as regras de negócio e fazer o orquestramento entre controllers e repositories. A camada service pode utilizar as entities para preparar um objeto para ser persistido na base ou para receber um objeto já persistido.

Repository - O componente repository é responsável por fazer a abstração para o acesso ao banco de dados, é nessa camada onde as operações necessárias a base são feitas. O repository faz o mapeamento da base para as entidades e a persistências das entidades para o banco de dados.

# Especificação de requisitos do sistema

Requisitos Funcionais

- Todo leilão cadastrado deve ter o nome do vendedor.

- Apenas usuários cadastrados deverão efetuar lances em um leilão.

- Para efetuar um lance em um leilão o usuário deve estar logado.

- Um usuário não logado poderá listar os leilões disponíveis.

- Todo lance efetuado em um leilão deve ter o nome do arrematante.

- Todo lance efetuado em um leilão deverá ter um valor igual ou maior que o lance mínimo estipulado pelo vendedor.

- Para efetuar um cadastro de usuário todos os campos devem estar preenchidos.

- Para efetuar o cadastro de um leilão todos os campos devem estar preenchidos.

- Para efetuar o cadastro de um novo usuário o sistema deve verificar se o usuário já está cadastrado.

- Para efetuar o login do usuário, o sistema deve verificar se o usuário está cadastrado.

- Requisitos Não Funcionais

- O sistema deverá guardar os dados de usuários de forma segura.

- O sistema não deverá exceder o tempo de resposta de 10ms pra cada requisição.

- O sistema deve rodar na maioria dos navegadores(Chrome, Edge, Mozilla, Brave ...).

- O sistema deve estar disponível na versão mobile.

- O sistema deverá ser desenvolvido em linguagem Java.

- O banco de dados deverá ser não relacional (mongoDB).

- O sistema deverá disponibilizar várias formas de pagamentos.

- O sistema deverá disponibilizar opções de entrega para leilões arrematados.

# Modelos do Sistema

Modelos gráficos do sistema que mostram os relacionamentos entre os componentes do sistema, o sistema e seu ambiente. Possíveis modelos são: modelos de objetos, modelos de fluxo de dados ou modelos semânticos de dados.

# Evolução do Sistema

Descreve os pressupostos fundamentais em que o sistema se baseia, bem como quaisquer mudanças previstas, em decorrência da evolução de hardware, de mudanças nas necessidades do usuário etc. Essa seção é útil para projetistas de sistema, pois pode ajudá-los a evitar decisões capazes de restringir possíveis mudanças futuras no sistema.

# Apêndices

Fornece informações detalhadas e específicas relacionadas à aplicação em desenvolvimento, além de descrições de hardware e banco de dados, entre outros. Os requisitos de hardware definem as configurações mínimas e ideais para o sistema. Requisitos de banco de dados definem a organização lógica dos dados usados pelo sistema e os relacionamentos entre esses dados.

# Índice

Vários índices podem ser incluídos no documento. Pode haver, além de um índice alfabético normal, um índice de diagramas, de funções, de tabelas, entre outros pertinentes.