**VISÃO DO PRODUTO**

Versão 1.0

25/072022

Alexandre Freire de Souza

Kelvem Carlos Fernandes Gomes

**HISTÓRICO**

| **Data** | **Versão** | **Descrição** | **Autor** |
| --- | --- | --- | --- |
| 28/07/22 | 1.0 | Definição de como será o sistema | Kelvem C. F. G. |
|  |  |  |  |
|  |  |  |  |

Sumário

[1. Introdução 4](#_gjdgxs)

[1.1 Escopo 4](#_1fob9te)

[1.2 Definições, Acrônimos e Abreviações 4](#_3znysh7)

[2.  Modelagem de Sistemas 4](#_2et92p0)

[2.1 Casos de Uso 4](#_tyjcwt)

[2.2 Modelo de Objetos 5](#_26in1rg)

[4. Modelo de Arquitetura REST 5](#_3dy6vkm)

[5. Padrões de Projeto 5](#_1t3h5sf)

[5. Frameworks 5](#_4d34og8)

[5.1 Spring Framework? 5](#_2s8eyo1)

[5.2 Hibernate 5](#_17dp8vu)

[6. Referências 5](#_3rdcrjn)

# 1. Introdução

O projeto Web 2 tem como Objetivo realizar o desenvolvimento de uma aplicação web usando o Java para web e outras ferramentas de desenvolvimento, com as opção de escolhas que ofereceram para minha dupla, optamos em escolher sistema de farmácia e drogarias, o intuito é que no final da cadeira, além de temos aprendido todas as competências para desenvolver uma aplicação Java para web e suas diversas ferramentas que ajudam no desenvolver de qualquer aplicação, é que possamos adquirir experiência para que possamos utilizar caso deseje trabalho para alguma empresa futuramente.

## Escopo

Bem, como sabemos, a função principal do sistema de uma farmácia ou drogaria é registrar as vendas, controlar o que foi vendido e arrecadado, fluxo de caixa e de estoque.

Sabemos que administrar uma farmácia vai muito além disso. Para atender um público cada vez mais exigente os processos tendem a aumentar.

Para este projeto, a função do sistema a ser criado é em relação ao cadastro de produtos voltados para área de farmácias e drogarias, tendo o intuito de melhorar o controle de estoque. Em relação ao uso das ferramentas para ajudar no desenvolvimento, principalmente a integração com o banco de dados, optamos por usar o MySql, onde ainda entrará em desenvolvimento.

*[Descrição do escopo deste documento e da área de negócio]*

## Definições, Acrônimos e Abreviações

BD – Banco de dados.

MVC – Model View Controller.

UML – Modelagem unificada.

IDE – Ambiente de desenvolvimento.

# 2.  Modelagem de Sistemas

Para a modelagem do nosso sistema vamos usar como base um crud. Nosso sistema deverá cadastrar um usuário, onde ele poderá fazer o login no sistema e tanto cadastrar os medicamentos como listar eles.

## 2.1 Casos de Uso

*[Descrição dos casos de uso e desenvolvimento do diagrama de casos de uso]*

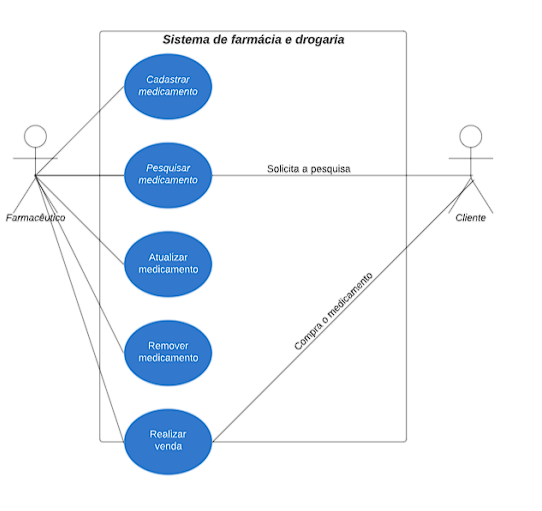
| Caso de Uso: Cadastrar Medicamentos |
| --- |
| **Descrição: O farmacêutico realizar o cadastro de medicamento no sistema** |
| **Cenário Principal:** |
| **1. Acesso do sistema.**  **2. Realização do cadastro dos medicamentos.**  **3. Finalizar ação.** |
| **Fluxo Alternativo:** |
| Sem fluxo alternativo |
| **Exceções: errar o cadastro do medicamento ou inserir dados inválidos.** |

| Caso de Uso: Pesquisar Medicamentos |
| --- |
| **Descrição: O farmacêutico realizar a busca pelo medicamento no sistema.** |
| **Cenário Principal:** |
| **1. Acesso do sistema.**  **2. Realização da busca pelo medicamento**  **3. Finalizar ação.** |
| **Fluxo Alternativo:** |
| Pesquisar pelo código do medicamento ou pelo nome. |
| **Exceções: erro de digitação, pesquisar medicamento não cadastrado, pesquisar medicamentos errados.** |

| Caso de Uso: Atualizar Medicamentos |
| --- |
| **Descrição: O farmacêutico atualiza os medicamentos registrados no sistema.** |
| **Cenário Principal:** |
| **1. Acesso do sistema.**  **2. Realização da busca pelos medicamentos.**  **3. Editar o medicamento.**  **4. Finalizar ação.** |
| **Fluxo Alternativo:** |
| Pesquisar pelo código do medicamento ou nome. |
| **Exceções: erro na digitação, pesquisar medicamento errado ou não registrado, erro na edição do medicamento.** |

| Caso de Uso: Remover Medicamento |
| --- |
| **Descrição: O farmacêutico realiza a remoção do medicamento no sistema** |
| **Cenário Principal:** |
| **1. Acesso do sistema.**  **2. Realização do cadastro dos medicamentos.**  **3. Remoção do medicamento registrado no sistema.**  **4. Finalizar ação.** |
| **Fluxo Alternativo:** |
| Pesquisar pelo código do medicamento ou pelo nome. |
| **Exceções: erro de digitação, pesquisar medicamento errado, medicamento não encontrado no sistema.** |

| Caso de Uso: Comprar Medicamentos |
| --- |
| **Descrição: O Cliente realiza a compra do medicamento.** |
| **Cenário Principal:** |
| **1. Solicitação da compra.**  **2. Farmacêutico realiza a pesquisa pelo medicamento no sistema**  **3. Informa os dados ao cliente.**  **4. Cliente efetua o pagamento.**  **5. Finalizar ação.** |
| **Fluxo Alternativo:** |
| Pesquisar pelo código do medicamento ou pelo nome, pagamento pelo cartão de crédito ou débito. |
| **Exceções: erro de digitação, pesquisa pelo medicamento errado, medicamento não encontrado no sistema, pagamento não efetuado, erro no pagamento.** |

**

## 2.2 Modelo de Objetos

## O modelo de objeto segue a abstração de um medicamento casual. O mesmo terá data de validade, data de fabricação, nome, tipo, valor e laboratório produtor. Assim, ao realizar o desenvolvimento o “farmacêutico” que utilizar o sistema poderá realizar o cadastro do medicamento seguindo e respeitando suas características. Segue abaixo o diagrama de classe UML:

# 

# 4. Modelo de Arquitetura REST

O desenvolvimento de sistemas distribuídos e sistemas web é uma atividade naturalmente complexa, pois envolve a criação de artefatos para o atendimento dos requisitos de negócio, bem como dos requisitos da aplicação como protocolos de comunicação e gerenciamento de transações distribuídas.

Todavia, o uso de boas práticas de programação aliado a um modelo arquitetural simplista e robusto podem reduzir drasticamente o tempo necessário para construir tais sistemas. Neste sentido, o REST (Representational State Transfer) apresenta-se como um modelo arquitetural poderoso e suficientemente maduro, capaz de atender requisitos complexos de integração sem agregar complexidade e excesso de acoplamento.

No modelo arquitetural REST, o protocolo HTTP (Hypertext Transfer Protocol) tem seus recursos amplamente explorados, firmando-se como um modelo de comunicação seguro, amplamente testado e, acima de tudo, padrão. Além do HTTP, outras tecnologias padrão de mercado são utilizadas, como XML (eXtensible Markup Language), JSON (Javascript Object Notation) e URI (Uniform Resource Identifier), por exemplo. Ao unir tais tecnologias amplamente conhecidas, o REST permite construir aplicações web distribuídas de forma rápida, robusta e com investimentos relativamente mais baixos quando comparado com outros modelos de arquitetura, como o SOA, por exemplo.

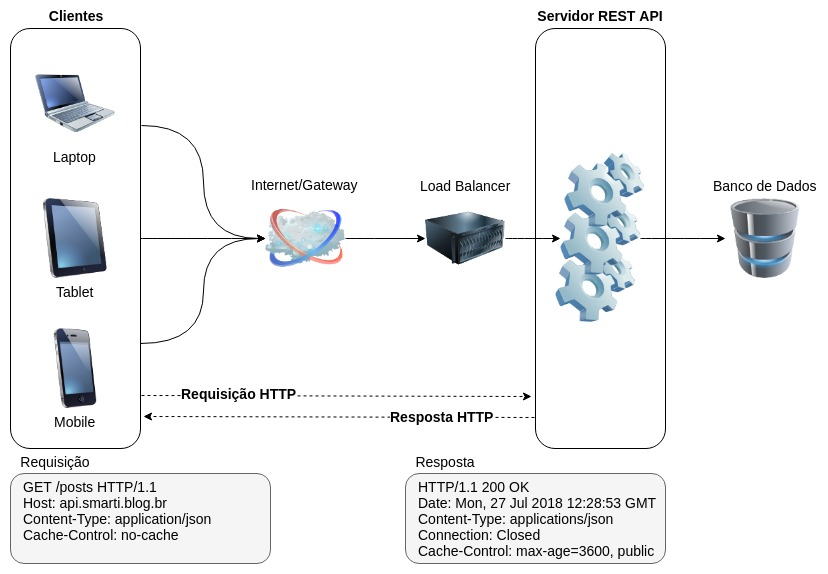


Imagem 4: Esquema de Funcionamento de Arquitetura REST.

# 5. Padrões de Projeto

Os padrões de projeto, como indica o nome, são padrões que devem ser seguidos no processo de desenvolvimento de um software. Este modelo é feito dentro da engenharia de software, mas passou a ser utilizado na grande maioria dos projetos quando se fala de tecnologia da informação.

Constantemente utilizado no desenvolvimento de softwares orientados a objetos, o design pattems tem como objetivo corrigir problemas encontrados no processo de desenvolvimento.

Neste projeto em questão será utilizado o padrão MVC (model, view e controller). Este padrão de projeto facilita a troca de informações entre a interface do usuário aos dados no banco, fazendo com que as respostas sejam mais rápidas e dinâmicas. O model será onde ficam as classes responsáveis por prover os objetos, juntamente com suas características e metodos. O controller funciona sendo intermediário entre a view e o model enviando. Já a view é responsável por apresentar as informações ao usuário.

# 5. Frameworks

## 5.1 Spring Framework?

## 5.2 Hibernate

# 6. Referências

TEDESCO, Kennedy. Padrões de projeto. treinaweb, 2017. Disponível em: https://www.treinaweb.com.br/blog/padroes-de-projeto-o-que-sao-e-o-que-resol vem

MARQUES, Luiz. Modelo de objeto. DCAA, 2001. Disponível em: https://www.dca.fee.unicamp.br/cursos/POOCPP/node9.html