

# Requisitos do Projeto

PI: Sistema Web Modularizável / Sistemas para Dispositivos Móveis | 4º Semestre / 5º Semestre

## Requisitos Funcionais

### APIs (Serviços)\*

ID	REQUISITO	REALIZADO?
01	API de pesquisa de produtos	
02	API de listagem de produtos	
03	API de listagem de categorias	
04	API de obtenção de produtos por categoria	
05	API de gerenciamento de conta (criar, manter)	
06	API de login (entrada com validação de acesso)	
07	API de adição ao carrinho de compras (adicionar, alterar quantidade)	
08	API de remoção do carrinho de compras (remover,limpar)	
09	API de fechamento de compra	
10	API de histórico de compras	

*\* Para ser considerado, o serviço deve estar plenamente funcional, atendendo ao que é solicitado, devolvendo e aceitando entrada de elementos JSON e possuir persistência no banco de dados.*

### Aplicação Cliente Android\*

ID	REQUISITO	REALIZADO?
11	Login e Logout	
12	Listagem de Produtos	
13	Pesquisa de Produtos	
14	Listagem de Produtos por Categoria	
15	Ver Detalhes do Produto	
16	Ver o Carrinho de Compras	
17	Adicionar um Produto ao Carrinho	
18	Editar Quantidade e Remover Produto no Carrinho	
19	Finalização da Compra	
20	Visualização de Histórico de Compras Pessoal	

*\*Para ser considerado realizado, os itens da aplicação móvel devem estar plenamente funcionais, atendendo ao que é solicitado, utilizando serviços para realização (quando indicado), terem interface gráfica bem construída e com boa usabilidade, validações de entrada e gerenciamento de erros*

# APIs

## 01 - API de pesquisa de produtos

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita realizar a pesquisa de produtos cadastrados na base de dados.
- A pesquisa deve ser feita pelo nome do produto.
- A API deve possuir uma variação que permita obter um determinado produto por ID.
- Os dados de entrada devem ser validados quanto a obrigatoriedade.

## 02 - API de listagem de produtos

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita realizar a listagem dos produtos cadastrados na base de dados.
- Caso desejado, pode-se utilizar paginação na listagem. Caso contrário, todos os produtos devem ser retornados.
- Os dados de entrada devem ser validados quanto a obrigatoriedade.

## 03 - API de listagem de categorias

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita listar as categorias de produtos cadastradas na base de dados.
- Caso desejado, pode-se utilizar paginação na listagem. Caso contrário, todas as categorias devem ser retornadas.

## 04 - API de obtenção de produtos por categoria

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita a listagem de todos os produtos de determinada categoria, informando o ID da categoria cujos produtos deseja-se obter.
- O serviço deve validar dados de entrada quanto a obrigatoriedade de elementos principais.

## 05 - API de gerenciamento de conta (criar, manter)

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita inserir, excluir, alterar e listar/procurar usuários numa base de dados.
- O serviço também deve permitir o envio de uma foto do usuário (usada no cadastro), seja o arquivo ou o endereço de uma imagem hospedada em outro local.
- Os dados de entrada devem ser validados quanto a obrigatoriedade.

## 06 - API de login (entrada com validação de acesso)

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita realizar o login da aplicação.
- O serviço também deve permitir enviar uma foto do usuário (usada no cadastro de usuário).
- Os dados de entrada devem ser validados quanto a obrigatoriedade.

## 07 - API de adição ao carrinho de compras (adicionar, alterar quantidade)

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita adicionar e atualizar a quantidade de itens de um carrinho de compras
- Os dados de entrada devem ser validados quanto a obrigatoriedade.

## 08 - API de remoção do carrinho de compras (remover, limpar)

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita remover e limpar um carrinho de compras
- Os dados de entrada devem ser validados quanto a obrigatoriedade.

## 09 - API de fechamento de compra

- Deve ser construído um serviço de API REST que trabalhe com dados em JSON e permita finalizar uma compra com base em um carrinho de compras.
- Os dados de entrada devem ser validados quanto a obrigatoriedade.

## 10 - API de histórico de compras

- Deve ser construído um serviço REST que trabalhe com dados em JSON e permita visualizar as compras já efetuadas por um usuário, mostrando a data da compra, o total, os produtos comprados e as quantidades.
- Os dados de entrada devem ser validados quanto a obrigatoriedade.

# Aplicação Cliente Android

## 11 – Login e Logout

- Deve ser possível entrar na aplicação móvel usando usuário e senha.
- Deve ser possível sair da aplicação e remover o login.
- Caso o usuário não tenha cadastro, deve ser possível realizar um para entrar.
- O login pode ser exibido antes do acesso a aplicação ou apenas caso o usuário vá finalizar uma compra (escolher a forma que preferir).
- O login deve ser feito consumindo a API da aplicação back-end.

## 12 - Listagem de Produtos

- A aplicação deve conter uma tela onde é possível ver uma listagem de produtos.
- Dados principais de produto e sua imagem devem ser exibidos na listagem.
- Ao clicar num produto da lista, a tela de detalhes deve ser exibida.
- A tela de listagem de produtos deve carregá-los utilizando o serviço correspondente.

## 13 - Pesquisa de Produtos

- A aplicação deve conter uma tela onde é possível pesquisar produtos por nome.
- Dados principais de produto e sua imagem devem ser exibidos na pesquisa.
- Ao clicar num produto da lista, a tela de detalhes deve ser exibida.
- A tela de pesquisa de produtos deve carregá-los utilizando o serviço correspondente.

## 14 - Listagem de Produtos por Categoria

- A aplicação deve conter uma tela onde é possível exibir todas as categorias de produtos e, quando clicadas, estas devem exibir todos os produtos daquela categoria.
- Dados principais de produto e sua imagem devem ser exibidos na listagem de produtos da categoria.
- Ao clicar num produto dos resultados da pesquisa, a tela de detalhes deve ser exibida.
- As telas de listagem de categorias e de produtos devem carregá-los utilizando o serviço correspondente.

## 15 - Detalhes do Produto

- Deve haver uma tela onde é possível visualizar todas as informações de um determinado produto.
- Os dados do produto devem ser carregados do serviço correspondente.
- Imagens do produtos deve ser exibidas e deve ser possível alternar entre elas.
- Deve ser possível adicionar o produto ao carrinho de compras.

## 16 - Ver o Carrinho de Compras

- Deve ser possível visualizar as informações do carrinho de compras numa tela específica.
- O carrinho de compras deve exibir o valor total da compra e os produtos no carrinho.
- Dados principais de produto e sua imagem devem ser exibidos no carrinho de compras.
- A listagem de produtos no carrinho deve carregar os dados necessários utilizando o serviço correspondente.

## 17 - Adicionar um Produto ao Carrinho

- Em todos os locais de listagem e detalhamento de produto, a aplicação deve permitir que o usuário o adicione ao carrinho de compras.
- A aplicação deve impedir a adição de mais de uma instância do mesmo produto no carrinho de compras (trabalhar com quantidades).
- A adição ao carrinho de compras deve utilizar o serviço correspondente.

## 18 - Editar Quantidade e Remover um Produto do Carrinho

- Na tela do carrinho de compras, deve ser possível editar as quantidades de todos os produtos no carrinho.
- Na tela do carrinho de compras, deve ser possível remover um produto do carrinho.
- A edição e remoção de produtos do carrinho de compras deve utilizar o serviço correspondente.

## **19 - Finalização da Compra**

- Na tela do carrinho de compras, deve ser possível finalizar a compra.
- A finalização da compra limpa o carrinho e grava os dados da compra nas tabelas respectivas do banco de dados usando o serviço correspondente.
- Uma mensagem deve ser exibida ao usuário informando que a compra foi finalizada.

## **20 - Visualização de Histórico de Compras Pessoal**

- Deve ser possível visualizar as informações de histórico das compras já efetuadas numa tela específica.
- A tela deve exibir os pedidos de compra já feitos, os produtos comprados, o valor total de cada pedido e a data de realização, agrupando os dados por pedido.