Descricao da pagina e-comerce

# Página de produtos

Produtos é formado por:

# Pagina de lista de produtos

Lista os produtos de uma determinada categoria (por exemplo, resistores). Esta página deve ser apresentada em forma de tabela como na Figura abaixo.

Cada produto deve conter seu próprio botão de comprar e área para entrar com a quantidade.

Estruturação da tabela de produtos

* As primeiras colunas devem ser preenchidas com características comuns a maioria dos produtos.
* O restante das colunas devem ser preenchidas com características específicas do tipo de produto

## Ações:

1. Comprar um único produto
2. Selecionar vários e Comprar vários
3. Ordenar por fabricante
4. Ordenar por Preço
5. Ordenar por disponibilidade
6. Ordenar por características específicas

Métodos:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Método | Descrição | Tipo | Categoria | Página |
| *prdGetCatgData* | Obter todos os produtos de uma determinada categoria | Object Array | Http |  |
| *prdgetCatg* | Obtem a categoria do produto. Deve guardar a |  |  |  |
| *prdGetId* |  | String |  |  |
| *prdGetIds* | Obter Ids de vários produtos | String Array |  |  |
| *prdRemKartItem* | Remover um item do carrinho |  |  | Carrinho |
| *prdGetQty* | Obter a quantidade de um item | Number int |  |  |
| *prdGetSubPrice* | Obter o subtotal de preços de um item | Number float |  | Carrinho |
| *prdGetTotalPrice* | Obter o Total de preço do carrinho |  |  | Carrinho |
| *prdKartCheckout* | Finalizar compra |  |  | Carrinho |
| *prdGetBySearch* | Obter produto através de busca |  |  |  |
| *prdOnPageReload* | Obter os dados quando o usuário dar um refresh na página |  |  |  |
| *prdGetKart* | Obter o carrinho de uma sessão anterior |  | cookies | Carrinho |
| *prdGetDetails* | Obtem os dados de um único produto através do seu ID |  |  | Detalhes |
| *prdGetByFilter* | Obtem os produtos usando os filtros |  | Http | Lista |
| *prdGetAll* | Obtém todos produtos de várias categorias ao mesmo tempo |  |  | Home |
| *prdStoreIdOnCkies* | Salva os Ids nos cookies |  |  |  |
| *prdGetBuy* | Obtem a compra |  |  |  |
| *prdSetCatgCkies* | Guarda um cookie com a categoria |  |  |  |

## Ações disparados por click:

As seguintes açoes serão disparadas por clicks de botões específicos

1. Comprar – Quando este botão for acionado, o produto será adicionado imediatamente ao carrinho de compras. O Toast do carrinho será exibido. Se o produto atual já estiver no carrinho, Um dialog será aberto. Este dialogo irá exibir a mensagem “PRODUTO JÄ SE ENCONTRA NO CARRINHO. O usuário perguntado CANCELAR. Se o usuário optar por adicionar, deve ser exibido

# Página de Detalhe de um único produto

# Carrinho de produtos

# Elementos secundários

# Toast na barra de ferramentas, exibido quando o usuário clica em comprar

Deve ser exibido todas as vezes que for adicionado um novo produto ou for acrescentado a quantidade de um produto já existente no carrinho.

## Implementação do Toast do Carrinho de compras

1. O Toast deve ter a estrutura apresentada na Figura abaixo. 
2. Deve ser exibido por mouseover no ícone do carrinho na barra de navegação
3. Deve ser exibido quando for adicionado um novo produto
4. Deve ser exibido quando for acrescentado a quantidade de um produto que já se encontra no carrinho
5. Deve permitir paginação
6. Exibir 5 produtos por pág
7. Links para acessar detalhes de cada um dos produtos
8. Botão para acessar o carrinho

# Produto duplicado

# Area na barra de tarefas que exibe Ícone de carrinho e a quantidade de produtos no carrin

# ho

# definição das funções

## prdKartRemItem(index)

Remove um item no carrinho. A função recebe como argumento o *index* da posição do item no carrinho. Esta função deve incluir a atualização do número de itens no carrinho. Esta eve ser atualizado a variável de serviço que guarda o número de itens no carrinho.

## prdKartCheckout()

Finaliza a compra.

1. Verificar se o usuário está logado. Se o usuário estiver logado, continuar para o procedimento. Se o usuário não estiver logado, abrir a pagina de login
2. Ao finalizar a compra, ir para a página que pergunta se gostaria de continuar comprando ou voltar para home
3. Limpar todos os dados do carrinho

# Pagina de listagem de produtos

A pagina de listagem de produtos será composto pelos seguintes elementos:

1. Listar todos os elementos da categoria selecionada no formato de tabela com os seus respectivos atributos
2. Permitir paginação
3. Permitir ordenação
4. Permitir seleção de vários produtos para comprar
5. Permitir seleção de até três produtos para comparação
6. Permitir comprar direto

Configurações:

1. Escolher o numero de produtos por pagina

# Usuário

# Login

## Redireccionamento para login

O usuário deve ser redirecionando para a página de login nas seguintes condições

* + 1. Tentar acessar a página “meus pedidos”
    2. Na finalização da compra
    3. Avaliar um produto

# Cadastro

Página para cadastro de usuários. Será composto por

# Página do usuário

Servico do usuário

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Método | Descrição | Tipo | Categoria | Página |
| *usr.login.getLogin* |  |  |  |  |
| *usr.login.getLogout* |  |  |  |  |
| *usr.login.clearForm* |  |  |  |  |
| *usr.cookies. put* |  |  |  |  |
| *usr.login.redirect* |  |  |  |  |
| *usr.cookies.get* |  |  |  |  |
| *Usr.cookies.clear* |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| *v* |  |  |  |  |

## Esqueci minha senha

Será implementado com passport

## Fazer uma pergunta ao vendedor

Este módulo vai permitir ao cliente fazer uma pergunta sobre um determinado produto. O cliente deve estar na página detalhes do produto para fazer a pergunta.

O vendedor deve receber um e-mail contendo a pergunta o nome do vendedor e os dados do produto.

As perguntas feitas e as respostas devem permanecer na página do produto.

’





