Atividades checkout.php

1. Como o sistema verifica se o carrinho contém itens antes de processar a compra?

R: O sistema irá verificar se a “$\_SESSION['cart']” não está vazia.

1. Como os itens do pedido são registrados no banco de dados?

R: Primeiro, insere um novo pedido na tabela pedidos, depois insere cada item do carrinho na tabela itens\_pedido.

1. O que acontece após o processamento da compra?

R: O carrinho é limpo, e o usuário é redirecionado para a página de confirmação (confirmacao\_pagamento.php).

1. Como o total do pedido é atualizado após o cálculo de todos os itens?

R: O total é calculado durante a inserção dos itens e depois atualizado na tabela pedidos usando o ID do pedido.

Atividades confirmacao\_pagamento.php

1. O que acontece com o carrinho após a confirmação do pagamento?

R: O carrinho é limpo utilizando unset($\_SESSION['cart']);.

1. Como a contagem de itens do carrinho é gerada na página de confirmação de pagamento?

R: A contagem é calculada somando as quantidades dos itens no carrinho com array\_sum(array\_column($\_SESSION['cart'], 'quantidade')).

1. Qual é o comportamento da página após o processamento do pagamento?

R: A página exibe uma mensagem de sucesso e oferece um link para voltar à página inicial.

1. Como a estrutura de navegação é gerada na página?

R: A estrutura de navegação é incluída através do arquivo header.php.

Atividades finalizar\_compra.php

1. O que ocorre se o carrinho estiver vazio ou o usuário não estiver logado?

R: Se o carrinho estiver vazio, o usuário é redirecionado para cart.php. Se o usuário não estiver logado, ele é redirecionado para login.php.

1. Como o sistema garante a integridade dos dados durante a finalização da compra?

R: O sistema utiliza transações com begin\_transaction(), commit() e rollback() para garantir que todas as operações no banco de dados sejam concluídas com sucesso antes de confirmar as mudanças. Em caso de erro, a transação é revertida para evitar dados inconsistentes.

1. Como o total do pedido é calculado e atualizado?

R: O total é calculado somando os preços dos itens multiplicados por suas quantidades durante a inserção dos itens no pedido. O total é então atualizado na tabela pedidos após a inserção dos itens.

1. O que acontece com os dados do carrinho após a conclusão da compra?

R: Os dados do carrinho são removidos da sessão com unset($\_SESSION['cart']).

Atividades gerar\_checkout.php

1. Qual é a função do arquivo gerar\_checkout.php?

R: O arquivo gerar\_checkout.php cria uma requisição de checkout para o PagSeguro com os dados do produto e redireciona o usuário para a página de pagamento do PagSeguro.

1. Como os dados do produto são enviados para o PagSeguro?

R: Os dados do produto são enviados em formato XML no corpo da requisição POST para a URL do PagSeguro, com as informações como ID do produto, nome, preço e quantidade.

1. Como o sistema redireciona o usuário para o pagamento do PagSeguro?

R: Após receber a resposta do PagSeguro, o sistema obtém o código do checkout da resposta XML e constrói a URL para a página de pagamento, redirecionando o usuário para essa URL usando header('Location: ' . $url\_pagamento).

1. Como o arquivo lida com erros na comunicação com o PagSeguro?

R: O arquivo não possui tratamento explícito para erros. Ele simplesmente obtém a resposta e processa, sem verificar se houve falha na requisição. Para um tratamento mais robusto, seria ideal verificar se $response é false e/ou validar o conteúdo da resposta antes de processar.

Atividades processa\_compra.php

1. Observe o seguinte código:

if (!isset($\_SESSION['usuario\_id'])) {

header('Location: login.php');

exit();

}

O que ele faz?

R: Esse código verifica se o usuário está logado verificando a presença da variável $\_SESSION['usuario\_id']. Se a variável não estiver definida (ou seja, o usuário não está logado), o usuário é redirecionado para a página de login (login.php) e o script é interrompido com exit().

2. Observe o seguinte código:

if (!isset($\_SESSION['cart']) || empty($\_SESSION['cart'])) {

header('Location: cart.php');

exit();

}

O que ele faz?

R: Esse código verifica se o carrinho de compras ($\_SESSION['cart']) está definido e não está vazio. Se o carrinho não estiver definido ou estiver vazio, o usuário é redirecionado para a página do carrinho (cart.php) e o script é interrompido com exit().

Atividades remover\_carrinho.php

1. Observe o seguinte código:  
     
   if (!isset($\_GET['id'])) { header('Location: cart.php'); exit(); }  
     
   O que ele faz? Explique.

R: Esse código verifica se o parâmetro id foi passado na URL (via $\_GET). Se id não estiver definido, o usuário é redirecionado para a página do carrinho (cart.php) e o script é interrompido com exit(). Isso garante que o processo de remoção só ocorra se um ID de produto for fornecido.

1. Observe o seguinte código:  
     
   if (isset($\_SESSION['cart'][$id])) { unset($\_SESSION['cart'][$id]); }  
     
   O que ele faz? Explique.

R: Esse código verifica se o item com o ID especificado ($id) está presente no carrinho ($\_SESSION['cart']). Se o item estiver no carrinho, ele é removido usando unset(). Isso efetivamente exclui o item do carrinho de compras.