**RELATÓRIO**

**Responsável: Ana Beatriz Loureiro**

Nesse relatório, encontram-se as propostas de melhorias no código do projeto de Supermercado.

**SUGESTÕES DA EQUIPE DE TESTE**

|  |
| --- |
| **SUGESTÃO #1: Nomeação adequada e legibilidade (Exemplo)** |
| **Arquivo:** CarrinhoDeCompras.java  **Código atual:**  Texto  Descrição gerada automaticamente com confiança baixa  **Sugestão de melhoria:**  Texto  Descrição gerada automaticamente com confiança média  **Justificativa**: os parâmetros “p” e “q” foram renomeados para “preço” e “quantidade” com o objetivo de deixar o código mais claro sobre o que está acontecendo. Além disso, o método foi simplificado para deixa-lo mais curto e melhorar a legibilidade. |

Abaixo, encontram-se os problemas identificados pela equipe de teste e as sugestões de melhorias propostas pelo responsável.

|  |
| --- |
| **SUGESTÃO #2: Métodos duplicados** |
| **Arquivo:** Cliente.java  **Código atual:**    **Sugestão de melhoria:**  package supermercado;  public class **Cliente** {  private String nome;  private String cpf;  private String email;  public Cliente(String nome, String cpf, String email) {  this.nome = nome;  this.cpf = cpf;  this.email = email;  }  public String getNome() {  return nome;  }  public void setNome(String nome) {  this.nome = nome;  }  public String getCpf() {  return cpf;  }  public void setCpf(String cpf) {  this.cpf = cpf;  }  public String getEmail() {  return email;  }  public void setEmail(String email) {  this.email = email;  }  }  **Justificativa**: Os métodos setNomeCliente, getNomeCliente, getCpfCliente, getEmailCliente são duplicados e redundantes e podem ser removidos. Mantendo apenas os métodos essenciais para a classe. Isso torna o código mais limpo e mais fácil de entender. |

|  |
| --- |
| **SUGESTÃO #3: Substituição de atributo por objeto** |
| **Arquivo:** Pedido.java  **Código atual:**    **Sugestão de melhoria:**  package supermercado;  public class Pedido {  private int numeroPedido;  private CarrinhoDeCompras carrinho;  private Cliente cliente;  public int getNumeroPedido() {  return this.numeroPedido;  }  public void setNumeroPedido(int numeroPedido) {  this.numeroPedido = numeroPedido;  }  public CarrinhoDeCompras getCarrinho() {  return carrinho;  }  public void setCarrinho(CarrinhoDeCompras carrinho) {  this.carrinho = carrinho;  }  public Cliente getCliente() {  return cliente;  }  public void setCliente(Cliente cliente) {  this.cliente = cliente;  }  public void fecharPedido() {  System.out.println("Numero do pedido: " + numeroPedido);  System.out.println("Cliente: " + cliente.getNome());  System.out.println("Total do pedido: " + this.carrinho.calcularTotal());  System.out.println("");  }  }  **Justificativa**: A classe Pedido não está utilizando corretamente as informações do cliente. Deveria armazenar o objeto Cliente em vez de strings separadas para nome, CPF e e-mail. |

|  |
| --- |
| **SUGESTÃO #4: Espaços em branco e recuo** |
| **Arquivo:** Supermercado.java  **Código atual:**  Interface gráfica do usuário, Texto, Aplicativo, Email  Descrição gerada automaticamente  **Sugestão de melhoria:**  package supermercado;  public class Supermercado {  // teste  public static void main(String[] args) {  // carrinho  CarrinhoDeCompras c = new CarrinhoDeCompras();  c.adicionarProduto("Arroz", 1, 3.99);  c.adicionarProduto("Filé de frango sassami", 2, 21.99);  c.calcularTotal();  //pedido  Pedido p = new Pedido();  p.setNumeroPedido(1);  p.setCarrinho(c);  p.fecharPedido();  }  }  **Justificativa**: Foi retirado os espaços em branco desnecessários. O código está mais compacto e sem os espaços em branco e recuo desnecessários, mantendo a mesma funcionalidade. |

|  |  |
| --- | --- |
| **SUGESTÃO #5: Comentários do código** | |
| **Arquivo:** Supermercado.java  **Código atual:**    **Sugestão de melhoria:**  package supermercado;  public class Supermercado {  // Método principal para teste  public static void main(String[] args) {  // Cria um carrinho de compras  CarrinhoDeCompras c = new CarrinhoDeCompras();  c.adicionarProduto("Arroz", 1, 3.99);  c.adicionarProduto("Filé de frango sassami", 2, 21.99);  // Calcula o total do carrinho  c.calcularTotal();  // Cria um pedido  Pedido p = new Pedido();  p.setNumeroPedido(1);  p.setCarrinho(c);  // Fecha o pedido  p.fecharPedido();  }  }  **Justificativa**: Os comentários foram ajustados para explicar o propósito de cada método e bloco de código, facilitando a compreensão do funcionamento do programa. Sem modificar a funcionalidade. | |
| **SUGESTÃO #6: Nomeação adequada** |
| **Arquivo:** Supermercado.java  **Código atual:**    **Sugestão de melhoria:**  public class Supermercado {  // Método principal para teste  public static void main(String[] args) {  // Cria um carrinho de compras  CarrinhoDeCompras carrinho = new CarrinhoDeCompras();  carrinho.adicionarProduto("Arroz", 1, 3.99);  carrinho.adicionarProduto("Filé de frango sassami", 2, 21.99);  // Calcula o total do carrinho  carrinho.calcularTotal();  // Cria um pedido  Pedido pedido = new Pedido();  pedido.setNumeroPedido(1);  pedido.setCarrinho(carrinho);  // Fecha o pedido  pedido.fecharPedido();  }  }  **Justificativa**: As variáveis e objetos estão nomeados de forma mais descritiva, facilitando a compreensão do código. |

**SUGESTÕES DO DESENVOLVEDOR**

Abaixo, encontram-se os problemas identificados e as sugestões de melhorias propostas pelo responsável.

|  |
| --- |
| **SUGESTÃO #7:** |
| **Arquivo:** Supermercado.java  **Código atual:**    **Sugestão de melhoria:**  public class Cliente {  private String nome;  private String cpf;  private String email;  public Cliente(String nome, String cpf, String email) {  this.nome = nome;  this.cpf = cpf;  this.email = email;  }  public String getNome() {  return nome;  }  public void setNome(String nome) {  this.nome = nome;  }  public String getCpf() {  return cpf;  }  public void setCpf(String cpf) {  this.cpf = cpf;  }  public String getEmail() {  return email;  }  public void setEmail(String email) {  this.email = email;  }  }  **Justificativa**: Esses métodos de acesso (getNome(), getCpf(), getEmail() e os métodos set) permitem que os atributos privados da classe Cliente sejam acessados e modificados de forma controlada. Isso ajuda a manter a encapsulação e a integridade dos dados da classe Cliente. |

|  |
| --- |
| **SUGESTÃO #8: Listar os produtos comprados final do pedido** |
| **Arquivo:** Supermercado.java  **Código atual:**  **Interface gráfica do usuário, Texto, Aplicativo, Email  Descrição gerada automaticamenteTexto  Descrição gerada automaticamente com confiança média**  **Sugestão de melhoria:**  package supermercado;  import java.util.ArrayList;  import java.util.List;  public class CarrinhoDeCompras {  private List<Produto> produtos;  public CarrinhoDeCompras() {  this.produtos = new ArrayList<>();  }  // Adiciona um produto ao carrinho  public void adicionarProduto(String nome, int quantidade, double precoUnitario) {  this.produtos.add(new Produto(nome, quantidade, precoUnitario));  }  // Calcula o total dos produtos no carrinho  public double calcularTotal() {  double total = 0;  for (Produto produto : produtos) {  total += produto.getQuantidade() \* produto.getPrecoUnitario();  }  return total;  }    // Retorna a lista de produtos do carrinho  public List<Produto> getProdutos() {  return produtos;  }  }package supermercado;  public class Pedido {  private int numeroPedido;  private CarrinhoDeCompras carrinho;  private Cliente cliente;    public int getNumeroPedido() {  return this.numeroPedido;  }  public Cliente getCliente() {  return this.cliente;  }  public void setCliente(Cliente cliente) {  this.cliente = cliente;  }    // Fecha o pedido, exibindo informações relevantes  public void fecharPedido() {  // Imprime o número do pedido  System.out.println("Número do pedido: " + numeroPedido);    // Imprime o nome do cliente  System.out.println("Cliente: " + cliente.getNome());    // Imprime os produtos do carrinho com suas descrições  System.out.println("Produtos:");  for (Produto produto : carrinho.getProdutos()) {  System.out.println("- " + produto.getNome() + ": " + produto.getQuantidade() + "x");  }    // Calcula e imprime o total do pedido formatado como moeda  System.out.printf("Total do pedido: R$ %.2f%n", this.carrinho.calcularTotal());    // Imprime uma linha de separação  System.out.println("=====================================");  // Adiciona descrição informando que o pedido foi fechado com sucesso  System.out.println("Pedido fechado com sucesso!");  }    public CarrinhoDeCompras getCarrinho() {  return carrinho;  }  public void setCarrinho(CarrinhoDeCompras carrinho) {  this.carrinho = carrinho;  }  public void setNumeroPedido(int numeroPedido) {  this.numeroPedido = numeroPedido;  }  }  **Justificativa**: Agora o método fecharPedido() exibe também as descrições dos produtos no carrinho, além de outros dados relevantes do pedido. |