

Vamos ao **Passo 5**, onde detalharemos a **integração do carrinho de compras** no aplicativo de loja de açaí. Esse passo envolve o gerenciamento de itens adicionados ao carrinho, a visualização desses itens, e a funcionalidade de finalizar a compra.

## Passo 5: Integração do Carrinho de Compras

### 5.1 Criando a Classe `CartManager`

O `CartManager` será responsável por gerenciar os itens do carrinho de compras. Ele armazenará os produtos que o usuário adiciona ao carrinho e fornecerá funções para manipular esses itens (adicionar, remover, e listar).

1. **Abra ou crie a classe** `CartManager.kt` (se não tiver sido feita no passo anterior):

kotlin



```
package com.example.acaiStore object CartManager { private val cartItems = mutableListOf<AcaiProduct>() // Adiciona um produto ao carrinho fun addToCart(product: AcaiProduct) { cartItems.add(product) } // Remove um produto do carrinho fun removeFromCart(product: AcaiProduct) { cartItems.remove(product) } // Retorna os itens do carrinho fun getCartItems(): List<AcaiProduct> = cartItems // Calcula o preço total do carrinho fun getTotalPrice(): Double = cartItems.sumOf { it.price } // Limpa o carrinho após a finalização da compra fun clearCart() { cartItems.clear() } }
```

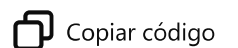
O `CartManager` é um objeto singleton que mantém uma lista de itens no carrinho e possui métodos para adicionar, remover e retornar a lista de produtos. Ele também calcula o total do carrinho e pode limpá-lo após a finalização da compra.

### 5.2 Adicionando ao Carrinho no `MainActivity`

Agora, vamos conectar a função de adicionar produtos ao carrinho no `MainActivity`. Quando o usuário clicar no botão “Adicionar”, o produto será adicionado ao carrinho.

1. **Abra o arquivo** `MainActivity.kt` e verifique se o código para adicionar ao carrinho está correto:

kotlin



```
package com.example.acaiStore import android.content.Intent import android.os.Bundle import android.widget.Toast import androidx.appcompat.app.AppCompatActivity import androidx.recyclerview.widget.LinearLayoutManager import
```

```

androidx.recyclerview.widget.RecyclerView import com.google.android.material.floatingactionbutton.FloatingActionButton class
MainActivity : AppCompatActivity() { private lateinit var recyclerView: RecyclerView private lateinit var acaiAdapter:
AcaiAdapter private lateinit var cartFab: FloatingActionButton private val products = listOf( AcaiProduct(1, "Açaí Tradicional",
"Açaí com banana e granola", 12.99, "https://example.com/acai1.jpg"), AcaiProduct(2, "Açaí com Morango", "Açaí com morango e
leite condensado", 15.99, "https://example.com/acai2.jpg"), AcaiProduct(3, "Açaí com Leite em Pó", "Açaí com leite em pó e mel",
14.99, "https://example.com/acai3.jpg") ) override fun onCreate(savedInstanceState: Bundle?) { super.onCreate(savedInstanceState)
setContentView(R.layout.activity_main) recyclerView = findViewById(R.id.recyclerView) cartFab = findViewById(R.id.fab_cart)
acaiAdapter = AcaiAdapter(products, onAddToCartClick = { product -> CartManager.addToCart(product) Toast.makeText(this,
"${product.name} adicionado ao carrinho!", Toast.LENGTH_SHORT).show() }, onItemClick = { product -> val intent = Intent(this,
ProductDetailsActivity::class.java) intent.putExtra("PRODUCT_ID", product.id) startActivity(intent) }) recyclerView.layoutManager
= LinearLayoutManager(this) recyclerView.adapter = acaiAdapter // Ação para navegar para o carrinho ao clicar no botão flutuante
cartFab.setOnClickListener { val intent = Intent(this, CartActivity::class.java) startActivity(intent) } }

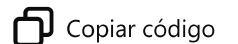
```

## 5.3 Tela de Carrinho de Compras

A tela do carrinho ( CartActivity ) exibirá os itens que foram adicionados ao carrinho e mostrará o valor total da compra. Também incluirá um botão para finalizar a compra.

### 1. Abra ou crie o arquivo CartActivity.kt :

kotlin



```

package com.example.acaiStore import android.os.Bundle import android.widget.Button import android.widget.TextView import
androidx.appcompat.app.AppCompatActivity import androidx.recyclerview.widget.LinearLayoutManager import
androidx.recyclerview.widget.RecyclerView import android.widget.Toast class CartActivity : AppCompatActivity() { private lateinit
var recyclerView: RecyclerView private lateinit var totalPriceTextView: TextView private lateinit var checkoutButton: Button
override fun onCreate(savedInstanceState: Bundle?) { super.onCreate(savedInstanceState) setContentView(R.layout.activity_cart)
recyclerView = findViewById(R.id.cartRecyclerView) totalPriceTextView = findViewById(R.id.totalPrice) checkoutButton =
findViewById(R.id.checkoutButton) // Obter itens do carrinho val cartItems = CartManager.getCartItems() // Configurar o
RecyclerView para exibir os itens do carrinho recyclerView.layoutManager = LinearLayoutManager(this) recyclerView.adapter =
AcaiAdapter(cartItems, {}, {}) // Atualizar o preço total val totalPrice = CartManager.getTotalPrice() totalPriceTextView.text =
"Total: R$ $totalPrice" // Ação do botão de finalizar compra checkoutButton.setOnClickListener { // Exemplo: exibir uma mensagem

```

```
de finalização e limpar o carrinho Toast.makeText(this, "Compra finalizada!", Toast.LENGTH_SHORT).show() CartManager.clearCart()  
// Limpar o carrinho finish() // Fechar a tela e retornar à anterior } }
```

## 5.4 Atualizando o Layout do Carrinho ( activity\_cart.xml )

### 1. Abra o arquivo activity\_cart.xml :

Aqui já definimos um layout para o carrinho no **Passo 3**. Vamos garantir que ele contém os elementos necessários:

xml



```
<?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout  
xmlns:android="http://schemas.android.com/apk/res/android" xmlns:app="http://schemas.android.com/apk/res-auto"  
android:layout_width="match_parent" android:layout_height="match_parent" android:padding="16dp"> <!-- RecyclerView para exibir os  
itens do carrinho --> <androidx.recyclerview.widget.RecyclerView android:id="@+id/cartRecyclerView" android:layout_width="0dp"  
android:layout_height="0dp" app:layout_constraintTop_toTopOf="parent"  
app:layout_constraintBottom_toTopOf="@+id/totalPriceContainer" app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintEnd_toEndOf="parent" android:padding="16dp" /> <!-- Contêiner para o preço total e botão de finalizar -->  
<LinearLayout android:id="@+id/totalPriceContainer" android:layout_width="0dp" android:layout_height="wrap_content"  
android:orientation="horizontal" android:gravity="center_vertical" app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintStart_toStartOf="parent" app:layout_constraintEnd_toEndOf="parent" android:padding="16dp"> <!-- Preço Total  
--> <TextView android:id="@+id/totalPrice" android:layout_width="wrap_content" android:layout_height="wrap_content"  
android:text="Total: R$ 0,00" android:textSize="20sp" android:textStyle="bold" android:layout_marginEnd="16dp" /> <!-- Botão de  
finalizar compra --> <Button android:id="@+id/checkoutButton" android:layout_width="wrap_content"  
android:layout_height="wrap_content" android:text="Finalizar Compra" android:backgroundTint="@color/purple_500"  
android:textColor="@android:color/white"/> </LinearLayout> </androidx.constraintlayout.widget.ConstraintLayout>
```

## 5.5 Atualizando o Adapter para o Carrinho

O `AcaiAdapter` que foi criado para exibir a lista de produtos será reutilizado para exibir os itens do carrinho. No caso do carrinho, não precisamos de botões "Adicionar ao Carrinho", pois os itens já estão no carrinho. Assim, o adapter pode ser reutilizado sem grandes alterações.

## 5.6 Testando a Funcionalidade do Carrinho

Agora que todas as funcionalidades de adicionar itens ao carrinho, visualizar o carrinho e finalizar a compra estão implementadas, você pode testar o fluxo do aplicativo:

1. Execute o aplicativo no emulador ou dispositivo físico.
2. Adicione produtos ao carrinho clicando no botão "Adicionar" na tela principal.
3. Abra o carrinho clicando no botão flutuante de carrinho de compras.
4. Verifique se os produtos adicionados aparecem no carrinho, e se o valor total é calculado corretamente.
5. Finalize a compra clicando no botão "Finalizar Compra" e verifique se o carrinho é limpo após a compra.

## Resumo

Com isso, a funcionalidade completa de carrinho de compras foi integrada ao aplicativo de venda de açaí. O `CartManager` gerencia os itens do carrinho, e o `CartActivity` exibe esses itens, permitindo que o usuário visualize o total e finalize a compra.