

Vamos ao **Passo 3**, onde detalharemos a criação da interface de usuário (Frontend com XML) no Android Studio para o aplicativo de venda de açaí. Esse passo envolve a criação dos layouts necessários para a exibição dos produtos, carrinho de compras e outros componentes visuais.


## Passo 3: Criação da Interface de Usuário (Frontend com XML)

### 3.1 Definindo o Layout Principal (activity\_main.xml)

O layout principal será a tela que exibe uma lista de produtos (açaí) que o usuário pode ver e adicionar ao carrinho.

1. **Abra o arquivo** `activity_main.xml` :
  - No painel esquerdo, vá até `res/layout/activity_main.xml` .
2. **Código para o** `activity_main.xml` :
  - Utilize um `RecyclerView` para exibir a lista de produtos e adicione um `FloatingActionButton` para navegar até o carrinho de compras.

xml

 Copiar código

```
<?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"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent" tools:context=".MainActivity"> <!-- RecyclerView
    para exibir a lista de produtos --> <androidx.recyclerview.widget.RecyclerView
    android:id="@+id/recyclerView" android:layout_width="0dp" android:layout_height="0dp"
    android:padding="16dp" app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent" /> <!-- Botão flutuante para acessar o
    carrinho de compras -->
    <com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/fab_cart" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:contentDescription="@string/cart"
    android:src="@drawable/ic_shopping_cart"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintBottom_toBottomOf="parent" app:tint="@color/white"
    app:backgroundTint="@color/purple_500" android:layout_margin="16dp"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

### 3.2 Criando o Layout Individual dos Produtos (item\_acai.xml)

Para exibir cada produto na lista, criaremos um layout específico chamado `item_acai.xml` .

1. **Crie o arquivo** `item_acai.xml` :

- No painel esquerdo, vá até `res/layout`, clique com o botão direito, e selecione **New > Layout Resource File**.
- Nomeie o arquivo como `item_acai.xml`.

## 2. Código para o `item_acai.xml`:

- O layout de cada item terá uma imagem do produto, nome, descrição e botão de adicionar ao carrinho.

xml

 Copiar código

```
<?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="wrap_content"
android:padding="8dp"> <!-- Imagem do produto --> <ImageView
android:id="@+id/productImage" android:layout_width="0dp"
android:layout_height="120dp" android:scaleType="centerCrop"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toStartOf="@+id/productDetails"
android:contentDescription="@string/product_image"/> <!-- Detalhes do produto (nome e
descrição) --> <LinearLayout android:id="@+id/productDetails"
android:orientation="vertical" android:layout_width="0dp"
android:layout_height="wrap_content" app:layout_constraintTop_toTopOf="parent"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintStart_toEndOf="@id/productImage"
app:layout_constraintEnd_toStartOf="@+id/addToCartButton"
android:layout_marginStart="8dp" android:gravity="center_vertical"> <!-- Nome do
produto --> <TextView android:id="@+id/productName"
android:layout_width="wrap_content" android:layout_height="wrap_content"
android:text="Açaí Tradicional" android:textSize="18sp" android:textStyle="bold"
android:textColor="@android:color/black"/> <!-- Descrição do produto --> <TextView
android:id="@+id/productDescription" android:layout_width="wrap_content"
android:layout_height="wrap_content" android:text="Açaí com banana e granola"
android:textSize="14sp" android:textColor="@android:color/darker_gray"/> <!-- Preço do
produto --> <TextView android:id="@+id/productPrice"
android:layout_width="wrap_content" android:layout_height="wrap_content"
android:text="R$ 12,99" android:textSize="16sp" android:textStyle="bold"
android:textColor="@android:color/holo_red_dark"/> </LinearLayout> <!-- Botão para
adicionar ao carrinho --> <Button android:id="@+id/addToCartButton"
android:layout_width="wrap_content" android:layout_height="wrap_content"
android:text="Adicionar" app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintBottom_toBottomOf="parent">
```


```
android:backgroundTint="@color/purple_500" android:textColor="@android:color/white"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

### 3.3 Criando a Tela de Detalhes do Produto (activity\_product\_details.xml)

Esta tela será aberta quando o usuário clicar em um produto para ver mais detalhes.

1. **Crie o arquivo** activity\_product\_details.xml :
  - Crie um novo arquivo de layout em res/layout e nomeie-o como activity\_product\_details.xml .
2. **Código para o** activity\_product\_details.xml :

xml

 Copiar código

```
<?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"> <!-- Imagem do produto --> <ImageView
android:id="@+id/productImageDetails" android:layout_width="0dp"
android:layout_height="200dp" android:scaleType="centerCrop"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:contentDescription="@string/product_image"/> <!-- Nome do produto -->
<TextView android:id="@+id/productNameDetails" android:layout_width="wrap_content"
android:layout_height="wrap_content" android:text="Açaí Tradicional"
android:textSize="24sp" android:textStyle="bold"
app:layout_constraintTop_toBottomOf="@id/productImageDetails"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent" android:layout_marginTop="16dp"/> <!--
Descrição do produto --> <TextView android:id="@+id/productDescriptionDetails"
android:layout_width="wrap_content" android:layout_height="wrap_content"
android:text="Açaí com banana e granola" android:textSize="18sp"
app:layout_constraintTop_toBottomOf="@id/productNameDetails"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent" android:layout_marginTop="8dp"/> <!-- Preço
do produto --> <TextView android:id="@+id/productPriceDetails"
android:layout_width="wrap_content" android:layout_height="wrap_content"
android:text="R$ 12,99" android:textSize="22sp" android:textStyle="bold"
app:layout_constraintTop_toBottomOf="@id/productDescriptionDetails"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent" android:layout_marginTop="16dp"/> <!-- Botão
para adicionar ao carrinho --> <Button android:id="@+id/addToCartButtonDetails"
android:layout_width="wrap_content" android:layout_height="wrap_content"
android:text="Adicionar ao Carrinho"
app:layout_constraintTop_toBottomOf="@id/productPriceDetails"
```

```

app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent" android:backgroundTint="@color/purple_500"
android:textColor="@android:color/white" android:layout_marginTop="24dp"/>
</androidx.constraintlayout.widget.ConstraintLayout>

```


### 3.4 Criando a Tela de Carrinho de Compras (activity\_cart.xml)

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

- Crie um novo arquivo de layout em res/layout e nomeie-o como activity\_cart.xml .

#### 2. Código para o activity\_cart.xml :

xml

 Copiar código

```

<?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>

```

## Resumo

Com os layouts principais

( activity\_main.xml , item\_acai.xml , activity\_product\_details.xml , activity\_cart.xml ) criados, a interface de usuário para o aplicativo de loja de venda de açaí está pronta. Agora você tem uma estrutura visual que permite aos usuários navegar pela lista de produtos, ver detalhes do produto e gerenciar o carrinho de compras.

Nos próximos passos, podemos conectar esses layouts com o backend Kotlin para carregar dados reais e gerenciar a interação do usuário.