

Aqui está o **Passo 2 detalhado** para a criação da interface principal do aplicativo **Tech Burguer** no Android Studio. Este passo inclui a definição do layout em XML e a configuração do `MainActivity.kt` para controlar a interface com Kotlin.

## 2. Estruturar a Interface Principal (MainActivity)

### Passo 2.1: Definir o Layout no arquivo `activity_main.xml`


Vamos criar um layout que inclui:

- Um `ImageView` para o logotipo do Tech Burguer.
- Um `RecyclerView` para listar os itens do menu de hambúrgueres.
- Botões para **ver o carrinho** e **finalizar o pedido**.

Estrutura do Layout `activity_main.xml` :

1. Navegue até o arquivo `res/layout/activity_main.xml` .
2. Substitua o conteúdo com o seguinte código para criar a interface principal:

xml

 Copiar código

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent" android:layout_height="match_parent"
    android:orientation="vertical" android:padding="16dp"> <!-- ImageView para o logotipo
--> <ImageView android:id="@+id/logoImageView" android:layout_width="wrap_content"
    android:layout_height="150dp" android:layout_gravity="center"
    android:src="@drawable/ic_burguer_logo" android:contentDescription="Tech Burguer Logo"
/> <!-- RecyclerView para exibir o menu de hambúrgueres -->
<androidx.recyclerview.widget.RecyclerView android:id="@+id/menuRecyclerView"
    android:layout_width="match_parent" android:layout_height="0dp"
    android:layout_weight="1" android:paddingTop="16dp" /> <!-- Botão para visualizar o
carrinho --> <Button android:id="@+id/viewCartButton"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:text="Ver Carrinho" android:layout_gravity="center_horizontal"
    android:layout_marginTop="16dp" /> <!-- Botão para finalizar o pedido --> <Button
    android:id="@+id/checkoutButton" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:text="Finalizar Pedido"
    android:layout_gravity="center_horizontal" android:layout_marginTop="8dp" />
</LinearLayout>
```

### Explicação:

- **ImageView:** Exibe o logotipo do Tech Burguer.
- **RecyclerView:** Exibe a lista de hambúrgueres disponíveis no menu.
- **Button (Ver Carrinho):** Permite ao usuário acessar o carrinho de compras.
- **Button (Finalizar Pedido):** Permite ao usuário finalizar o pedido e ir para a tela de checkout.

### Passo 2.2: Criar o Layout de Item do RecyclerView ( `burger_item.xml` )

O **RecyclerView** exibirá cada hambúrguer em um item de layout individual. Para isso, criamos um novo arquivo de layout para o item:

1. Crie um novo arquivo de layout em **res/layout** chamado `burger_item.xml`.
2. Adicione o seguinte código para representar cada item do menu de hambúrgueres:

xml

 Copiar código

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent" android:layout_height="wrap_content"
    android:orientation="horizontal" android:padding="16dp"
    android:gravity="center_vertical"> <!-- ImageView para a imagem do hambúrguer -->
    <ImageView android:id="@+id/burgerImageView" android:layout_width="100dp"
    android:layout_height="100dp" android:src="@drawable/ic_burger"
    android:contentDescription="Imagem do Hambúrguer" /> <!-- Layout para o texto de nome,
    descrição e preço --> <LinearLayout android:layout_width="0dp"
    android:layout_height="wrap_content" android:layout_weight="1"
    android:orientation="vertical" android:layout_marginStart="16dp"> <!-- Nome do
    hambúrguer --> <TextView android:id="@+id/burgerNameTextView"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:text="Nome do Burger" android:textSize="18sp" android:textStyle="bold" /> <!--
    - Descrição do hambúrguer --> <TextView android:id="@+id/burgerDescriptionTextView"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:text="Descrição do Burger" android:textSize="14sp"
    android:textColor="#757575" android:layout_marginTop="4dp" /> <!-- Preço do hambúrguer
    --> <TextView android:id="@+id/burgerPriceTextView"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:text="R$ 00,00" android:textSize="16sp" android:textColor="#FF9800"
    android:layout_marginTop="8dp" /> </LinearLayout> </LinearLayout>
```

### Explicação:

- **ImageView:** Exibe a imagem do hambúrguer.
- **TextViews:** Exibem o nome, descrição e preço do hambúrguer.

### Passo 2.3: Definir o Código Kotlin no `MainActivity.kt`

Agora, vamos configurar a lógica para exibir a lista de hambúrgueres no `RecyclerView` e associar os botões para o carrinho e checkout.

1. Abra o arquivo **MainActivity.kt**.
2. Substitua o conteúdo pelo seguinte código:

kotlin

 Copiar código

```
package com.example.techburger import android.os.Bundle import
androidx.appcompat.app.AppCompatActivity import
androidx.recyclerview.widget.LinearLayoutManager import
androidx.recyclerview.widget.RecyclerView import android.widget.Button import
```

```

android.widget.Toast class MainActivity : AppCompatActivity() { private lateinit var
burgerAdapter: BurgerAdapter private lateinit var menuRecyclerView: RecyclerView
private lateinit var viewCartButton: Button private lateinit var checkoutButton:
Button override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState) setContentView(R.layout.activity_main) //
Inicializar RecyclerView menuRecyclerView = findViewById(R.id.menuRecyclerView)
menuRecyclerView.layoutManager = LinearLayoutManager(this) // Inicializar botões
viewCartButton = findViewById(R.id.viewCartButton) checkoutButton =
findViewById(R.id.checkoutButton) // Criar lista de hambúrgueres val burgerList =
listOf( Burger("Cheeseburger", "Pão, carne, queijo, tomate, alface", 19.99,
R.drawable.ic_burger), Burger("Bacon Burger", "Pão, carne, bacon, queijo, cebola",
22.99, R.drawable.ic_burger), Burger("Veggie Burger", "Pão, vegetais, queijo, alface,
tomate", 18.99, R.drawable.ic_burger) ) // Configurar Adapter burgerAdapter =
BurgerAdapter(burgerList) { burger -> // Lógica ao clicar em um item do menu
Toast.makeText(this, "Adicionado: ${burger.name}", Toast.LENGTH_SHORT).show() }
menuRecyclerView.adapter = burgerAdapter // Configurar botões
viewCartButton.setOnClickListener { Toast.makeText(this, "Abrir carrinho",
Toast.LENGTH_SHORT).show() } checkoutButton.setOnClickListener { Toast.makeText(this,
"Finalizar pedido", Toast.LENGTH_SHORT).show() } } }

```

### Explicação:

- **RecyclerView:** Configura o RecyclerView para exibir a lista de hambúrgueres.
- **Botões:** Define as ações para os botões "Ver Carrinho" e "Finalizar Pedido".
- **BurgerAdapter:** Configura o adaptador do RecyclerView para exibir cada item de hambúrguer (veremos o `BurgerAdapter` no próximo passo).

### Passo 2.4: Criar o Adaptador para o RecyclerView (BurgerAdapter)

1. Crie um novo arquivo Kotlin chamado `BurgerAdapter.kt` na pasta `java/com.example.techburger`.
2. Adicione o seguinte código para definir o adaptador do RecyclerView:

kotlin

 Copiar código

```

package com.example.techburger import android.view.LayoutInflater import
android.view.View import android.view.ViewGroup import android.widget.ImageView import
android.widget.TextView import androidx.recyclerview.widget.RecyclerView class
BurgerAdapter(private val burgerList: List<Burger>, private val clickListener:
(Burger) -> Unit) : RecyclerView.Adapter<BurgerAdapter.BurgerViewHolder>() { class
BurgerViewHolder(view: View) : RecyclerView.ViewHolder(view) { val burgerName:
TextView = view.findViewById(R.id.burgerNameTextView) val burgerDescription: TextView
= view.findViewById(R.id.burgerDescriptionTextView) val burgerPrice: TextView =
view.findViewById(R.id.burgerPriceTextView) val burgerImage: ImageView =
view.findViewById(R.id.burgerImageView) } override fun onCreateViewHolder(parent:
ViewGroup, viewType: Int): BurgerViewHolder { val view =
LayoutInflater.from(parent.context).inflate(R.layout.burger_item, parent, false)
return BurgerViewHolder(view) } override fun onBindViewHolder(holder:

```

```
BurgerViewHolder, position: Int) { val burger = burgerList[position]
holder.burgerName.text = burger.name holder.burgerDescription.text =
burger.description holder.burgerPrice.text = "R$ ${burger.price}"
holder.burgerImage.setImageResource(burger.imageResId)
holder.itemView.setOnClickListener { clickListener(burger) } } override fun
getItemCount(): Int { return burgerList.size } }
```

### Explicação:

- **BurgerAdapter:** O adaptador preenche cada item do RecyclerView com os dados dos hambúrgueres e lida com cliques nos itens.

Com isso, o **Passo 2** está completo. A interface principal foi configurada, incluindo o layout em XML e a lógica em Kotlin para manipular a lista de hambúrgueres.