Implementação ZooApp

QXD0276 - Desenvolvimento de Software para Dispositivos Móveis

Universidade Federal do Ceará - *Campus* Quixadá

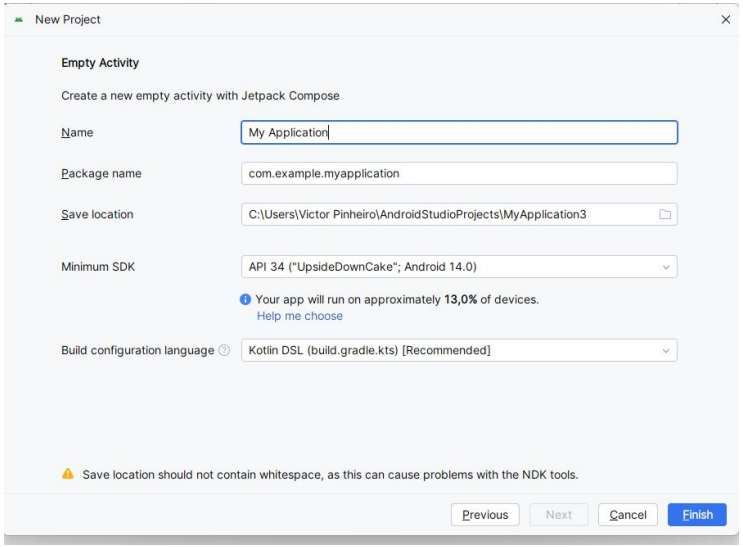
Prof. Francisco Victor da Silva Pinheiro
victorpinheiro@ufc.br

Agenda

- Passo 1: Criar o Projeto no Android Studio
- Passo 2: Configuração do Gradle
- Passo 3: Estrutura do Projeto
- Passo 4: Criar o Modelo
- Passo 5: Criar os Componentes
- Passo 6: Criar Telas
- Passo 7: Configurar Navegação
- Passo 8: Configurar MainActivity
- Execução do ZooApp

Passo 1: Criar o Projeto no Android Studio

- Abra o Android Studio.
- Crie um novo projeto:
  - Selecione a opção **Empty Compose Activity**.
  - Nomeie o projeto como **ZooApp**.
  - Configure o pacote como, por exemplo, ***com.example.zooapp***.
  - Certifique-se de que a linguagem está definida como Kotlin e que a opção Use Jetpack Compose está marcada.
- Configure o mínimo SDK suportado (recomendo API 21 ou superior).



Passo 2: Configuração do Gradle

- No arquivo build.gradle do módulo, verifique se as dependências do Compose estão atualizadas:

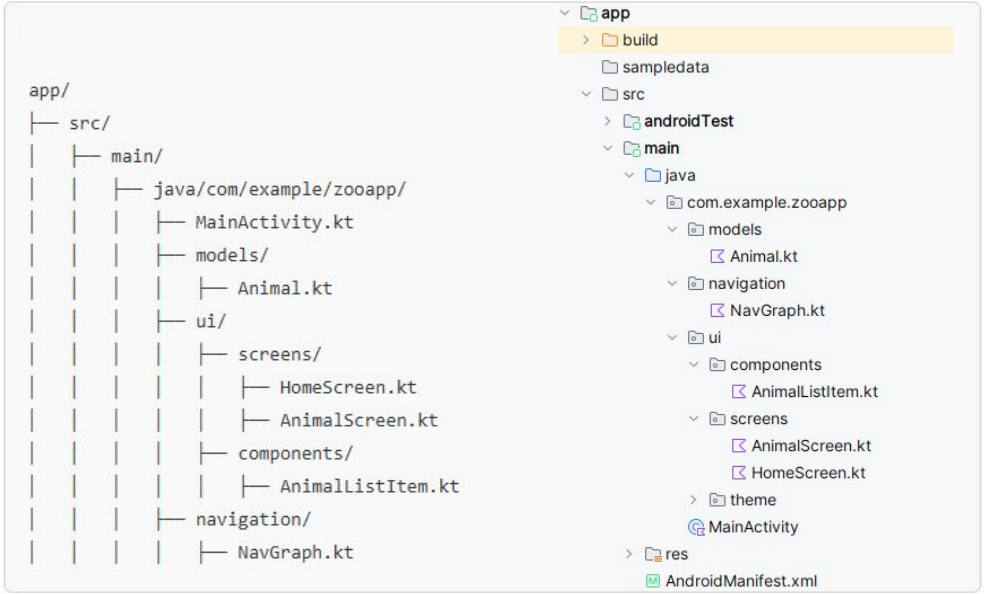
```
implementation ("androidx.compose.ui:ui:1.5.0")
implementation ("androidx.compose.material3:material3:1.1.1")
implementation ("androidx.navigation:navigation-compose:2.5.3")
implementation ("androidx.lifecycle:lifecycle-runtime-compose:2.6.1")
implementation ("androidx.activity:activity-compose:1.7.2")
```

- Sincronize o projeto para baixar as dependências.

Passo 3: Estrutura do Projeto

- Organize o projeto para separação lógica entre telas, componentes, modelos e navegação:

```
app/
├── src/
│   ├── main/
│   │   ├── java/com/example/zooapp/
│   │   │   ├── MainActivity.kt
│   │   │   ├── models/
│   │   │   │   ├── Animal.kt
│   │   │   ├── ui/
│   │   │   │   ├── screens/
│   │   │   │   │   ├── HomeScreen.kt
│   │   │   │   │   ├── AnimalScreen.kt
│   │   │   │   ├── components/
│   │   │   │   │   ├── AnimalListItem.kt
│   │   │   ├── navigation/
│   │   │   │   ├── NavGraph.kt
```



Passo 4: Criar o Modelo

- Crie um arquivo Animal.kt na pasta models/:

```
package com.example.zooapp.models

import com.example.zooapp.R


data class Animal (
    val id: Int,
    val name: String,
    val species: String,
    val imageRes: Int,
    val description: String,
    val curiosities: String,
    var isFavorite: Boolean = false
)
```

Passo 4: Criar o Modelo

- Adicione uma lista mock de animais:

```
val animalList = listOf(
    Animal(
        id = 1,
        name = "Dog",
        species = "Canis lupus familiaris",
        imageRes = R.drawable.cat,
        description = "O cão é um dos animais mais antigos domesticados pelo homem.",
        curiosities = "Os cães têm um olfato cerca de 40 vezes mais potente que o dos humanos."
    ),
    Animal(
        id = 2,
        name = "Cat",
        species = "Felis catus",
        imageRes = R.drawable.cat,
        description = "O gato doméstico é conhecido por sua agilidade e independência.",
        curiosities = "Gatos passam cerca de 70% do dia dormindo."
    ),
)
```

Passo 5: Criar os Componentes - Parte 1

- Arquivo: AnimalListItem.kt na pasta components/:

```
package com.example.zooapp.ui.components

import androidx.compose.foundation.Image
import androidx.compose.foundation.layout.*
import androidx.compose.foundation.shape.CircleShape
import androidx.compose.material3.*
import androidx.compose.runtime.Composable
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.draw.clip
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.unit.dp
import com.example.zooapp.models.Animal
```

Passo 5: Criar os Componentes - Parte 2

```
@Composable
fun AnimalListItem(animal: Animal, onAnimalSelected: (Animal) ->
Unit) {
    Card(
        modifier = Modifier
            .fillMaxWidth()
            .padding( 8.dp),
        elevation = CardDefaults.cardElevation( 4.dp)
    ) {
        Column(
            modifier = Modifier
                .fillMaxWidth()
                .padding( 16.dp)
        ) {
            Row(
                verticalAlignment = Alignment.CenterVertically
            ) {
                Image(
                    painter = painterResource( id =
animal.imageRes),
```

```
        contentDescription = "${animal.name} Image",
                    modifier = Modifier
                        .size( 80.dp)
                        .clip(CircleShape)
                )
                Spacer( modifier = Modifier.width( 16.dp))
                Column {
                    Text(text = animal.name, style =
MaterialTheme.typography.titleMedium)
                    Text( text = animal.species, style =
MaterialTheme.typography.bodySmall)
                }
            }
            Spacer(modifier = Modifier.height( 16.dp))
            Text(
                text = animal.description,
                style = MaterialTheme.typography.bodyMedium
            )
            Spacer( modifier = Modifier.height( 8.dp))
```

Passo 5: Criar os Componentes - Parte 2

```
            Text(
                text = "Curiosidade: ${animal.curiosities}",
                style = MaterialTheme.typography.bodySmall,
                color = MaterialTheme.colorScheme.secondary
            )
            Spacer(modifier = Modifier.height(8.dp))
            Button(onClick = { onAnimalSelected(animal) }) {
                Text("Mais sobre ${animal.name}")
            }
        }
    }
}
```

Passo 6: Criar Telas - Parte 1

- Arquivo: HomeScreen.kt na pasta screens/:

```
package com.example.zooapp.ui.screens

import androidx.compose.foundation.layout.*
import androidx.compose.material3.*
import androidx.compose.runtime.*
import androidx.compose.ui. Modifier
import androidx.compose.ui.unit.dp
import androidx.compose.foundation.lazy.LazyColumn
import androidx.compose.foundation.lazy.items
import com.example.zooapp.models. Animal
import com.example.zooapp.models. animalList
import com.example.zooapp.ui.components.AnimalListItem

@Composable
fun HomeScreen(onAnimalSelected : (Animal) -> Unit) {
    var searchQuery by remember { mutableStateOf("") }
    val filteredAnimals = remember(searchQuery) {
        animalList.filter { it.name.contains(searchQuery, ignoreCase = true) }
    }
```

Passo 6: Criar Telas - Parte 2

- Arquivo: HomeScreen.kt na pasta screens/:

```
    Column {
        TextField(
            value = searchQuery,
            onValueChange = { searchQuery = it },
            label = { Text("Pesquisar") },
            modifier = Modifier
                .fillMaxWidth()
                .padding(8.dp)
        )
        LazyColumn(
            verticalArrangement = Arrangement.spacedBy(8.dp),
            modifier = Modifier.padding(horizontal = 8.dp)
        ) {
            items(filteredAnimals) { animal ->
                AnimalListItem(animal, onAnimalSelected)
            }
        }
    }
}
```

Passo 6: Criar Telas - Parte 1

- Arquivo: AnimalScreen.kt na pasta screens/:

```
package com.example.zooapp.ui.screens

import androidx.compose.foundation.Image
import androidx.compose.foundation.layout.*
import androidx.compose.foundation.shapeCircleShape
import androidx.compose.material3.*
import androidx.compose.runtimeComposable
import androidx.compose.uiAlignment
import androidx.compose.uiModifier
import androidx.compose.ui.draw.clip
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.unit.dp
import com.example.zooapp.modelsAnimal

@ExperimentalMaterial3Api
@Composable
fun AnimalScreen(animal: Animal) {
   Scaffold(
       topBar = {
           TopAppBar(title = { Text(animal.name) })
       }
   ) { paddingValues ->
```

Passo 6: Criar Telas - Parte 2

- Arquivo: AnimalScreen.kt na pasta screens/:

```
Column(
            modifier = Modifier
                .fillMaxSize()
                .padding(paddingValues)
                .padding(16.dp),
            horizontalAlignment =Alignment.CenterHorizontally
        ) {
            Image(
                painter = painterResource(id = animal.imageRes),
                contentDescription ="${animal.name} Image",
                modifier = Modifier
                    .size(200.dp)
                    .clip(CircleShape)
            )
            Spacer(modifier = Modifier.height(16.dp))
            Text(
                text = animal.species,
                style = MaterialTheme.typography.titleMedium
            )
            Spacer(modifier = Modifier.height(16.dp))
            Text(
                text = animal.description,
                style = MaterialTheme.typography.bodyMedium
            )
```

Passo 6: Criar Telas - Parte 2

- Arquivo: AnimalScreen.kt na pasta screens/:

```
            Spacer(modifier = Modifier.height(16.dp))
            Text(
                text = "Curiosidade: ${animal.curiosities}",
                style = MaterialTheme.typography.bodySmall,
                color = MaterialTheme.colorScheme.secondary
            )
        }
    }
}
```

Passo 7: Configurar Navegação

- Arquivo: NavGraph.kt na pasta navigation/:

```
package com.example.zooapp.navigation

import androidx.compose.material3.ExperimentalMaterial3Api
import androidx.compose.runtime.Composable
import androidx.navigation.compose.NavHost
import androidx.navigation.compose.composable
import androidx.navigation.compose.rememberNavController
import com.example.zooapp.ui.screens.AnimalScreen
import com.example.zooapp.ui.screens.HomeScreen
import com.example.zooapp.models.animalList

@ExperimentalMaterial3Api
@Composable
fun NavGraph() {
   val navController = rememberNavController()

   NavHost(navController = navController, startDestination = "home") {
       composable("home") {
           HomeScreen(onAnimalSelected = { animal ->
               navController.navigate("animal/${animal.name}")
           })
       }
       composable("animal/{animal}") { backStackEntry ->
           val animalName = backStackEntry.arguments?.getString("animal")
           val selectedAnimal = animalList.first { it.name == animalName }
           AnimalScreen(selectedAnimal)
       }
   }
}
```

Passo 8: Configurar MainActivity

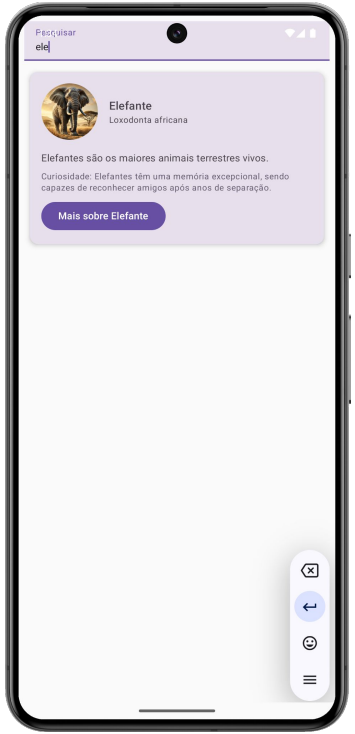
- Arquivo: MainActivity.kt:

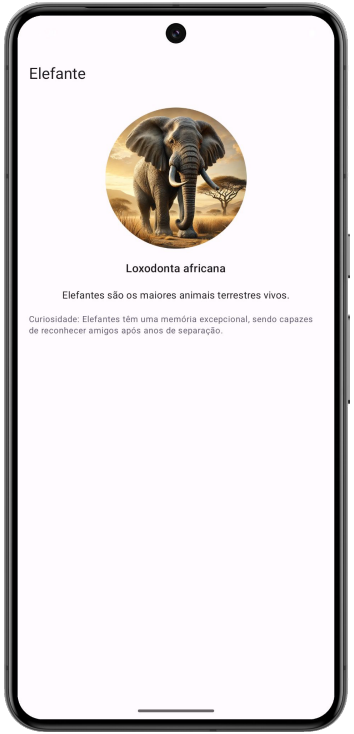
```
package com.example.zooapp

import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.material3.ExperimentalMaterial3Api
import com.example.zooapp.navigation.NavGraph

@ExperimentalMaterial3Api
class MainActivity : ComponentActivity() {
   override fun onCreate(savedInstanceState: Bundle?) {
       super.onCreate(savedInstanceState)
       setContent {
           NavGraph()
       }
   }
}
```

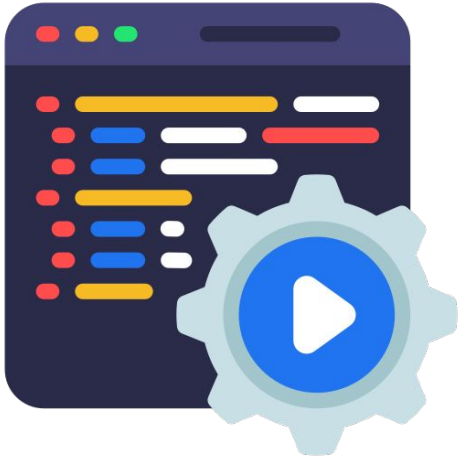
Execução do ZooApp





Referências

- https://developer.android.com/
- https://developer.android.com/courses/fundamentals-training/
- https://flutter.dev/



Obrigado!

Dúvidas?

Universidade Federal do Ceará - *Campus* Quixadá

Prof. Francisco Victor da Silva Pinheiro
victorpinheiro@ufc.br