



Search

Write



m

Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)



# Android Compose — Making Splash Screen with Lottie

Ridvan Özcan · [Follow](#)

4 min read · Mar 17, 2024

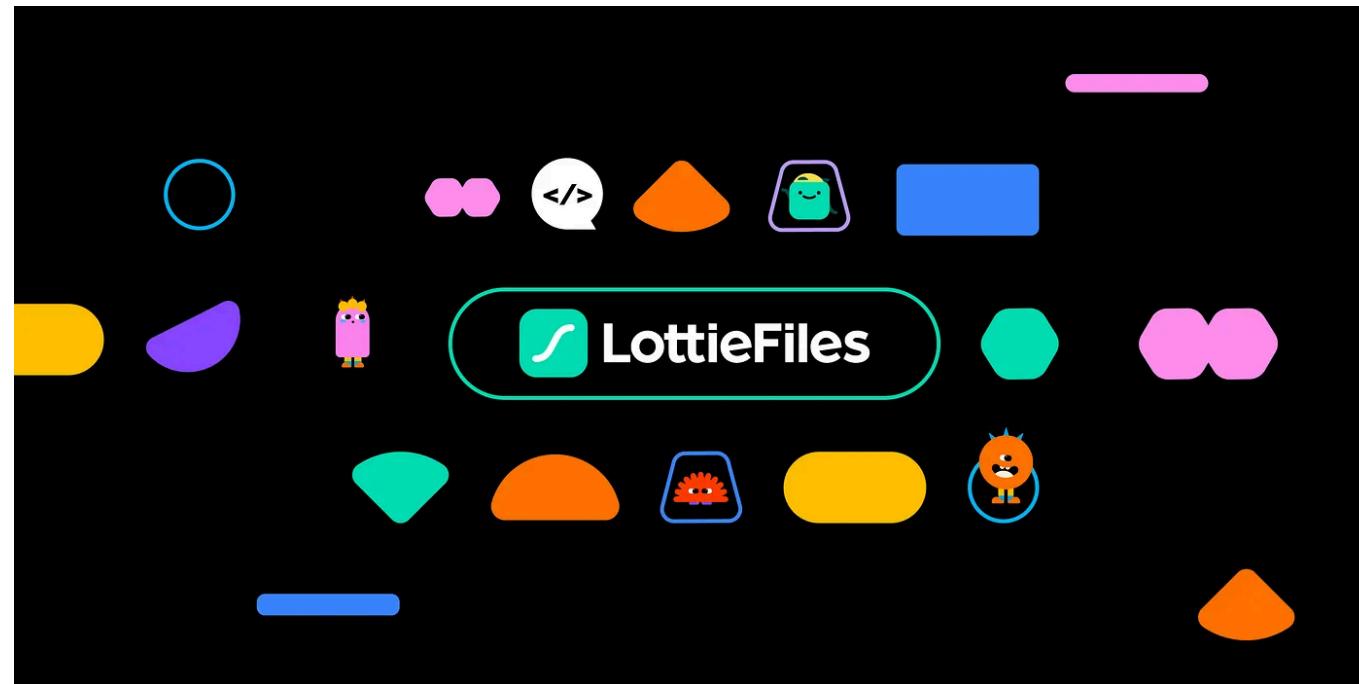


65



...

The Android Splash Screen serves as the initial interface that greets users upon launching an application. In this tutorial, we'll craft a dynamic splash screen utilizing [Lottie](#), a powerful animation library for Android, in conjunction with Navigation-Compose



<https://lottiefiles.com/>

Let us add Navigation-Compose and Lottie to your app level *build.gradle* file.

```
dependencies{
    ...
    implementation 'com.airbnb.android:lottie-compose:6.3.0'
    implementation 'androidx.navigation:navigation-compose:2.7.7'
```

```
...  
}
```

We will have to set the background colour and the system bar colours in order for use to play our animations nicely. We will add the below properties to the `themes.xml` file

```
<style name="Theme.AnimatedSplashScreen" parent="android:Theme.Material.Light.NoActionBar">  
    <item name="android:statusBarColor">@color/black</item>  
    <item name="android:windowBackground">@color/black</item>  
</style>
```

Let's kick things off by defining a sealed class named `Destination` to manage the routes within our navigation graph. Given this is a basic example, we'll keep it simple with just two screens: the splash screen and the home screen.

```
sealed class Screen(val route: String) {  
    object Splash : Screen("splash_screen")  
    object Home : Screen("home_screen")  
}
```

Here, we've established a sealed class called `Screen`. Inside it, we've defined two objects, `Splash` and `Home`, each representing a screen in our app. Additionally, we've assigned a `route` property to each object, which will be used for navigation purposes.

Next, let's configure our navigation graph. We'll create a new Kotlin file named `SetupNavGraph.kt` to manage the navigation graph for our screens.

```
@Composable
fun SetupNavGraph(navController: NavHostController) {
    NavHost(
        navController = navController,
        startDestination = Screen.Splash.route
    ) {
        composable(route = Screen.Splash.route) {
            SplashScreen(navController = navController)
        }
        composable(route = Screen.Home.route) {
            Box(
                modifier = Modifier
                    .fillMaxSize()
                    .background(Color.White),
                contentAlignment = Alignment.Center
            ) {
                Text(text = "Hello World", color = Color.Black)
            }
        }
    }
}
```

```
    }  
}
```

In this code, we've defined a composable function `SetupNavGraph` responsible for setting up our navigation graph. We're utilizing Jetpack Compose's navigation library to achieve this.

The `NavHost` composable sets up a navigation host with a specified `NavController` and a starting destination. Inside `NavHost`, we've defined two destinations using the `composable` composable. Each destination is associated with a route from our `Destination` sealed class.

To link the navigation graph to the main activity, we'll call the `SetupNavGraph` function inside our main activity.

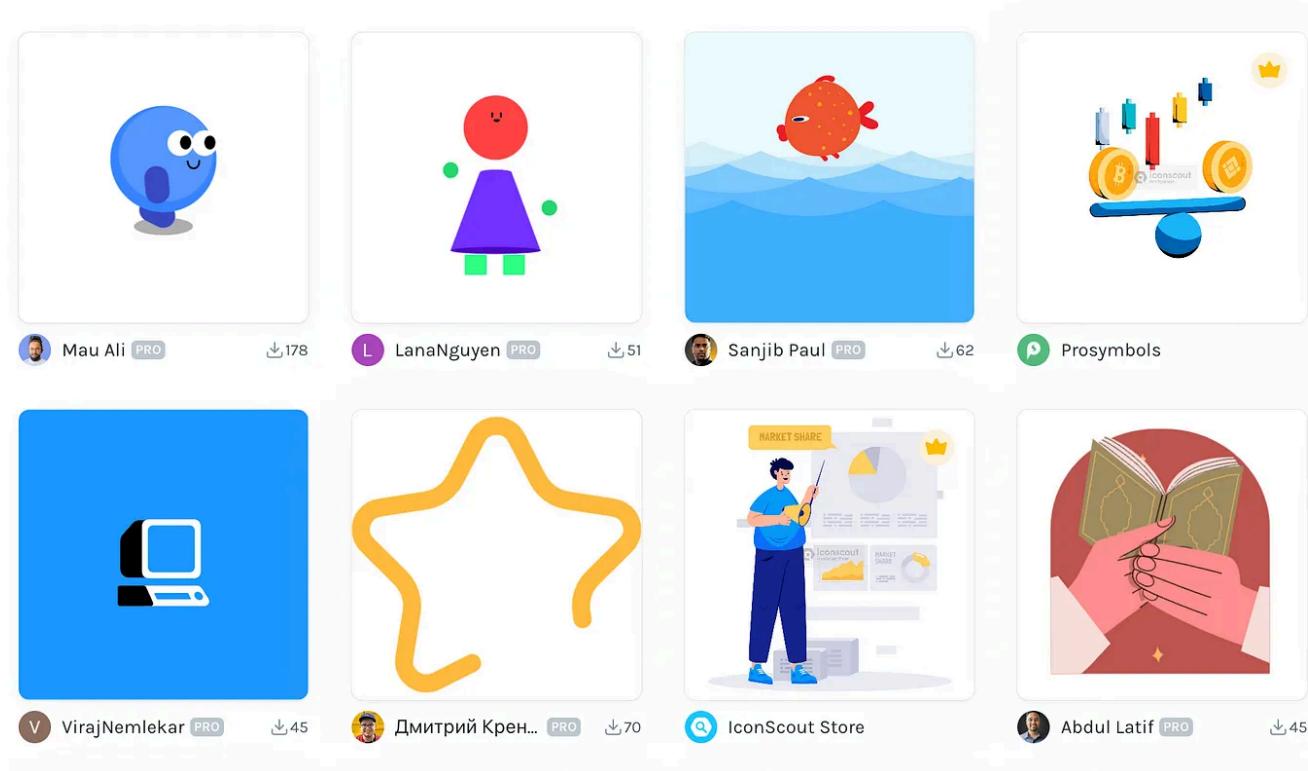
```
class MainActivity : ComponentActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContent {  
            AnimatedSplashScreenTheme {  
                val navController = rememberNavController()  
                SetupNavGraph(navController = navController)  
            }  
        }  
    }  
}
```

```
        }  
    }  
}
```

In this code, we've set the content of the activity to a composable function that initializes the navigation graph. We've also used `rememberNavController()` to create a navigation controller that will manage navigation within the app.

Now, our main activity is set up to display the navigation graph defined in `SetupNavGraph.kt`, which handles navigation between the splash screen and the home screen.

Now download [lottie](#) and let us build the screen which will play our animation and then navigate to the home screen. We will simply add a Lottie view and load a lottie file in the view.



<https://lottiefiles.com/featured>

Now we can create a SplashScreen for managing ui and logic.

```
@Composable
fun SplashScreen(navController: NavHostController) {
    Box(
        modifier = Modifier
            .fillMaxSize()
            .background(MaterialTheme.colors.primary)
    ) {
```

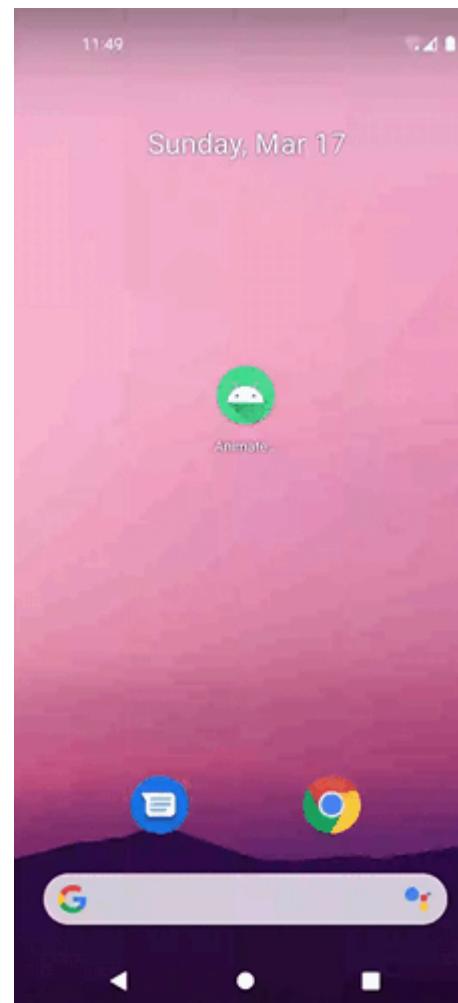
```
val composition by rememberLottieComposition(LottieCompositionSpec.RawRe  
val logoAnimationState =  
    animateLottieCompositionAsState(composition = composition)  
LottieAnimation(  
    composition = composition,  
    progress = { logoAnimationState.progress }  
)  
if (logoAnimationState.isAtEnd && logoAnimationState.isPlaying) {  
    navController.navigate(Screen.Home.route)  
}  
}  
}
```

The provided code sets up a composition for utilization within our Lottie view. Direct assignment of the “logo.lottie” file from the raw folder in the resources is facilitated. Subsequently, a state variable is initialized to monitor the animation progression within the composable, enabling subsequent checks for animation completion.

Finally, integration of the `LottieAnimation` view is accomplished by associating the composed components and progress. Upon state transition to “playing” and reaching the final frame, seamless navigation to the next composable is executed, thus achieving the desired functionality.

For the complete, operational code, please refer to the following link:  
<https://github.com/ridvanozcan/AnimatedSplashScreen>

## Preview



If you have any questions or feedback, please feel free to contact me. Best regards, Rıdvan Özcan.

### **ridvanozcan — Overview**

You can't perform that action at this time. You signed in with another tab or window. You signed out in another tab or...

[github.com](https://github.com/ridvanozcan)

<https://www.linkedin.com/in/ridvanozcan/>

<https://www.instagram.com/mr.softwareengineer/>

#kotlin #kotlindeveloper #android #androiddeveloper  
#softwaredevelopment #androiddevelopment #kotlinandroid #developers  
#developercommunity #softwareengineer #googledevs #googlegroups  
#androidgroups #googleandroid #gradle #gradleandroid #community  
#linkedinforcreators #creators #linkedin #linkedinlearning #linkedinfamily  
#linkedinconnections #animated #splashscreen #lottie

[Lottie](#)[Splash Screen](#)[Android](#)[Compose](#)[Lottie Animation](#)

## Written by Rıdvan Özcan

107 Followers

Android Developer | [Sahibinden.com](https://sahibinden.com)

[Follow](#)

---

## More from Rıdvan Özcan



Google Maps



Ridvan Özcan

## How to Use Google Maps in Jetpack Compose: Step-by-Step...

In this tutorial, we will guide you through the integration of Jetpack Compose and Google...

6 min read · Mar 27, 2024



57



Ridvan Özcan

## Google Play Store'un uygulama yayinallyamak için 20 testçi politika...

Google Play Store şimdidiye kadar her türlü geliştiriciye hızlı bir şekilde uygulama...

3 min read · Dec 25, 2023



36



Ridvan Özcan



Ridvan Özcan

## Android Activity Lifecycle Quick Interview

Understand the Activity Lifecycle

4 min read · Apr 17, 2024



28



...

## Setup Charles Proxy for Android Emulators / MacOS

The information provided in the help section of Charles Proxy for Android emulators is...

6 min read · Apr 19, 2023



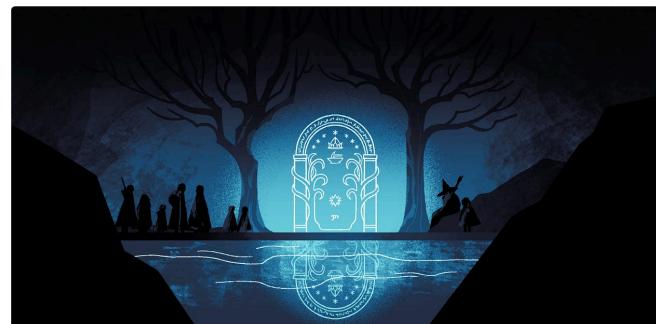
104



...

See all from Rıdvan Özcan

## Recommended from Medium



sed magna elit....

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit. In  
sed magna elit....

Lorem ipsum dolor sit amet,



Samed Harman

## Android Compose—Making Splash Screen with SplashScreen...

Hi, In this article we are going to make this splash screen example in Android Jetpack...

5 min read · Dec 25, 2023



Francesc Vilarino Guell

## Shared Element Transitions in Jetpack Compose

Introduction

7 min read · Apr 21, 2024



## Lists



### Staff Picks

634 stories · 934 saves

Jetpack Compose

## Setting Up Material Theme Color Schemes in Jetpack Compose

Medium

<https://medium.com/@rowaido.game>





Michihiro Iwasaki

## Setting Up Material Theme Color Schemes in Jetpack Compose

Simplifying Color Decisions with Material Design

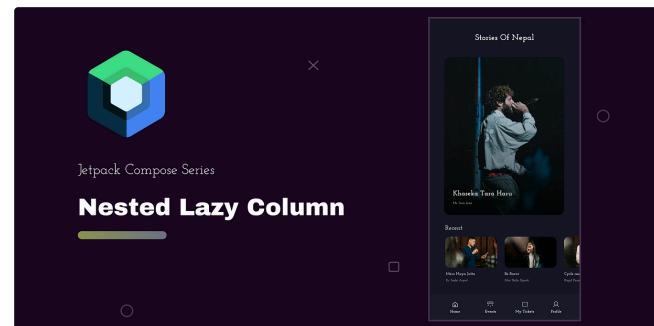
6 min read · Nov 28, 2023



62



1



Narayan Panthi in Kt. Academy

## Nested LazyColumn in Jetpack Compose

When displaying groups of elements, we generally use columns and rows. But when it...

7 min read · Apr 17, 2024



169



4



Tdkms

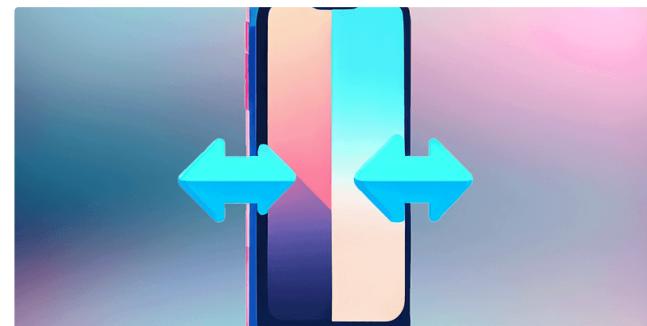
## Theme switching in Android Jetpack Compose

In this tutorial, I will teach you how to switch between themes in Android Jetpack...

3 min read · Feb 26, 2024



51



Tomáš Repčík

## Jetpack Compose and Screen Transition Animations

The accompanist compose animations got deprecated and migrated to the standard...

5 min read · Nov 4, 2023



146



[See more recommendations](#)

---

[Help](#) [Status](#) [About](#) [Careers](#) [Blog](#) [Privacy](#) [Terms](#) [Text to speech](#) [Teams](#)