

# **A REDUX INSPIRE PODCAST APP WITH DYNAMIC THEMES FOR ANDROID**

## **TEAM MATES:**

- 1.KAVIPRIYAN.D**
- 2.VISHWA SUBASH.S**
- 3.NARASIMMA MOORTHY.M**
- 4.VISHAL BOOPATHI.R**
- 5.SHIVA PRAKASH.J**

# DESCRIPTION

---

**A Reduction-Inspired Podcast App for Android, designed to offer a streamlined, minimalist podcast experience, focuses on simplicity, clarity, and efficiency. The app embraces the "reduction" philosophy by removing unnecessary features and clutter, providing a distraction-free interface to let users focus purely on their listening experience.**

# Key Features

---

- **Dynamic Themes:** Tailored to your environment and time of day, the app offers customizable themes that shift with natural lighting, mood, or personal preferences. Whether you prefer a dark theme for night listening or a light, airy design for daytime use, the app adapts to your needs.
- **Intuitive Interface:** A clean, easy-to-navigate UI with minimal buttons and distractions. Users can effortlessly browse, subscribe, and play podcasts with simple gestures.
- **Focus Mode:** A special feature that removes all non-essential UI elements, leaving only the playback controls and episode details, ensuring complete focus on the content.
- **Smart Recommendations:** The app learns your listening habits over time and suggests new podcasts based on your interests without overwhelming you with endless choices.
- **Offline Listening:** Download episodes for offline listening and automatically manage storage, so you can enjoy your podcasts anytime, anywhere.
- **Sleep Timer & Playback Speed:** Adjust settings to your comfort with sleep timer functionality and customizable playback speed.
- **Podcasts in a Clean List:** No more messy lists! Podcasts are organized simply, based on latest episodes or favorite categories.

# program

```
package com.example.podcastplayer

import android.content.Context
import android.media.MediaPlayer
import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.foundation.BorderStroke
import androidx.compose.foundation.Image
import androidx.compose.foundation.layout.*
import androidx.compose.foundation.rememberScrollState
import androidx.compose.foundation.verticalScroll
import androidx.compose.material.*
import androidx.compose.runtime.*
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.text.style.TextAlign
```

```
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.em
import androidx.compose.ui.unit.sp
import com.example.podcastplayer.ui.theme.PodcastPlayerTheme
```

```
class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            PodcastPlayerTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background

                ) {
                    playAudio(this)

                }
            }
        }
    }
}
```

```
}  
}  
}
```

```
@Composable  
fun playAudio(context: Context) {
```

```
Column(modifier = Modifier.fillMaxSize()) {
```

```
Column(horizontalAlignment = Alignment.CenterHorizontally, verticalArrangement =  
Arrangement.Center) {  
    Text(text = "PODCAST",  
modifier = Modifier.fillMaxWidth(),  
    textAlign = TextAlign.Center,  
    color = Color(0xFF6a3ef9),  
    fontWeight = FontWeight.Bold,  
    fontSize = 36.sp,  
style = MaterialTheme.typography.h1,
```

**letterSpacing = 0.1*em***

**)  
}**

**Column(modifier = Modifier  
    *.fillMaxSize()*  
    *.verticalScroll(rememberScrollState())*) {**

**Card(  
        elevation = 12*.dp*,  
border = *BorderStroke*(1*.dp*, Color.Magenta),  
        modifier = Modifier  
            *.padding*(16*.dp*)  
            *.fillMaxWidth()*  
            *.height*(250*.dp*)  
    )  
    {**

**val mp: MediaPlayer = MediaPlayer.create(context, R.raw.*audio*)**



```
Column(  
    modifier = Modifier.fillMaxSize(),  
    horizontalAlignment = Alignment.CenterHorizontally  
    ) {
```

```
    Image(  
        painter = painterResource(id = R.drawable.img),  
        contentDescription = null,  
        modifier = Modifier  
            .height(150.dp)  
            .width(200.dp),  
    )
```

```
    Text(  
        text = "GaurGopalDas Returns To TRS - Life, MonkhooD & Spirituality",  
        textAlign = TextAlign.Center,  
        modifier = Modifier.padding(start = 20.dp, end = 20.dp)  
    )
```



**Row() {**

```
IconButton(onClick = { mp.start() }, modifier = Modifier.size(35.dp)) {  
    Icon(  
        painter = painterResource(id = R.drawable.play),  
        contentDescription = ""  
    )  
}
```

```
IconButton(onClick = { mp.pause() }, modifier = Modifier.size(35.dp)) {  
    Icon(  
        painter = painterResource(id = R.drawable.pause),  
        contentDescription = ""  
    )  
}  
  
}  
  
}
```

```
        Card(  
            elevation = 12.dp,  
            border = BorderStroke(1.dp, Color.Magenta),  
            modifier = Modifier  
                .padding(16.dp)  
                .fillMaxWidth()  
                .height(250.dp)  
        )  
    {  
val mp: MediaPlayer = MediaPlayer.create(context, R.raw.audio_1)  
  
        Column(  
            modifier = Modifier.fillMaxSize(),  
            horizontalAlignment = Alignment.CenterHorizontally  
        ) {  
  
            Image(  

```

```
painter = painterResource(id = R.drawable.img_1),
        contentDescription = null,
        modifier = Modifier
            .height(150.dp)
            .width(200.dp)
    )

    Text(
text = "Haunted Houses, Evil Spirits & The Paranormal Explained | Sarbajeet Mohanty",
        textAlign = TextAlign.Center,
        modifier = Modifier.padding(start = 20.dp, end = 20.dp)
    )

    Row() {

        IconButton(onClick = { mp.start() }, modifier = Modifier.size(35.dp)) {
            Icon(
                painter = painterResource(id = R.drawable.play),
                contentDescription = ""
            )
        }
    }
```

```
        Text(
text = "Kaali Mata ki kahani - Black Magic & Aghoris ft. Dr Vineet Aggarwal",
        textAlign = TextAlign.Center,
        modifier = Modifier.padding(start = 20.dp, end = 20.dp)
        )

    Row() {

        IconButton(onClick = { mp.start() }, modifier = Modifier.size(35.dp)) {
            Icon(
                painter = painterResource(id = R.drawable.play),
                contentDescription = ""
            )
        }

        IconButton(onClick = { mp.pause() }, modifier = Modifier.size(35.dp)) {
            Icon(
                painter = painterResource(id = R.drawable.pause),
                contentDescription = ""
            )
        }

    }

}

    Card(
        elevation = 12.dp,
        border = BorderStroke(1.dp, Color.Magenta),
        modifier = Modifier
            .padding(16.dp)
            .fillMaxWidth()
            .height(250.dp)
    )
}

val mp: MediaPlayer = MediaPlayer.create(context, R.raw.audio_3)
```

```
}  
}  
  
}
```

```
        Card(  
            elevation = 12.dp,  
border = BorderStroke(1.dp, Color.Magenta),  
            modifier = Modifier  
                .padding(16.dp)  
                .fillMaxWidth()  
                .height(250.dp)  
        )  
    {  
val mp: MediaPlayer = MediaPlayer.create(context, R.raw.audio_2)
```

```
        Column(  
            modifier = Modifier.fillMaxSize(),  
horizontalAlignment = Alignment.CenterHorizontally  
  
        ) {
```

```
            Image(  
painter = painterResource(id = R.drawable.img_2),  
            contentDescription = null,  
            modifier = Modifier  
                .height(150.dp)  
                .width(200.dp)  
            )
```

```
        Column(
            modifier = Modifier.fillMaxSize(),
            horizontalAlignment = Alignment.CenterHorizontally
        ) {

            Image(
                painter = painterResource(id = R.drawable.img_3),
                contentDescription = null,
                modifier = Modifier
                    .height(150.dp)
                    .width(200.dp),

            )

            Text(
                text = "Tantra Explained Simply | Rajarshi Nandy - Mata, Bhairav & Kamakhya Devi",
                textAlign = TextAlign.Center,
                modifier = Modifier.padding(start = 20.dp, end = 20.dp)
            )
            Row() {

                IconButton(onClick = { mp.start() }, modifier = Modifier.size(35.dp)) {
                    Icon(
                        painter = painterResource(id = R.drawable.play),
                        contentDescription = ""
                    )
                }

                IconButton(onClick = { mp.pause() }, modifier = Modifier.size(35.dp)) {
                    Icon(
                        painter = painterResource(id = R.drawable.pause),
                        contentDescription = ""
                    )
                }
            }
        }
    }
```

```
        Card(
            elevation = 12.dp,
            border = BorderStroke(1.dp, Color.Magenta),
            modifier = Modifier
                .padding(16.dp)
                .fillMaxWidth()
                .height(250.dp)
        )
    }

    val mp: MediaPlayer = MediaPlayer.create(context, R.raw.audio_4)

    Column(
        modifier = Modifier.fillMaxSize(),
        horizontalAlignment = Alignment.CenterHorizontally
    ) {

        Image(
            painter = painterResource(id = R.drawable.img_4),
            contentDescription = null,
            modifier = Modifier
                .height(150.dp)
                .width(200.dp),

        )

        Text(
            text = "Complete Story Of Shri Krishna - Explained In 20 Minutes",
            textAlign = TextAlign.Center,
            modifier = Modifier.padding(start = 20.dp, end = 20.dp)
        )

        Row() {

            IconButton(onClick = { mp.start() }, modifier = Modifier.size(35.dp)) {
                Icon(
                    painter = painterResource(id = R.drawable.play),
                    contentDescription = ""
                )
            }
        }
    }
}
```



```
        IconButton(onClick = { mp.pause() }, modifier = Modifier.size(35.dp)) {
            Icon(
                painter = painterResource(id = R.drawable.pause),
                contentDescription = ""
            )
        }
    }
}

    Card(
        elevation = 12.dp,
        border = BorderStroke(1.dp, Color.Magenta),
        modifier = Modifier
            .padding(16.dp)
            .fillMaxWidth()
            .height(250.dp)
    )
    {
        val mp: MediaPlayer = MediaPlayer.create(context, R.raw.audio_5)

        Column(
            modifier = Modifier.fillMaxSize(),
            horizontalAlignment = Alignment.CenterHorizontally
        ) {
            Image(
                painter = painterResource(id = R.drawable.img_5),
                contentDescription = null,
                modifier = Modifier
                    .height(150.dp)
                    .width(200.dp),
            )

            Text(
                text = "Mahabharat Ki Poori Kahaani - Arjun, Shri Krishna & Yuddh - Ami Ganatra ",
```

```
        textAlign = TextAlign.Center,  
        modifier = Modifier.padding(start = 20.dp, end = 20.dp)  
    )  
    Row() {
```

```
        IconButton(onClick = { mp.start() }, modifier = Modifier.size(35.dp)) {  
            Icon(  
                painter = painterResource(id = R.drawable.play),  
                contentDescription = ""  
            )  
        }
```

```
        IconButton(onClick = { mp.pause() }, modifier = Modifier.size(35.dp)) {  
            Icon(  
                painter = painterResource(id = R.drawable.pause),  
                contentDescription = ""  
            )  
        }  
    }  
}  
}
```

# output

