Logotipo, nombre de la empresa

El contenido generado por IA puede ser incorrecto.

**INSTITUTO TECNOLÓGICO NACIONAL DE MÉXICO**

**INSTITUTO TECNOLÓGICO DE CIUDAD MADERO**

**Tu primera app para Android**

**LUIS FERNANDO HERNANDEZ MERCADO 20070531**

**HUGO ANDRE MARTINEZ GALVAN 20070545**

**Unidad 1: Tu primera app para Android**

**Ruta de aprendizaje 3**

package com.example.happybirthday

Crea un diseño básico

// Importaciones necesarias para la actividad principal y componentes de Jetpack Compose

import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.layout.\*

import androidx.compose.material3.MaterialTheme

import androidx.compose.material3.Surface

import androidx.compose.material3.Text

import androidx.compose.runtime.Composable

import androidx.compose.ui.Alignment

import androidx.compose.ui.Modifier

import androidx.compose.ui.text.style.TextAlign

import androidx.compose.ui.tooling.preview.Preview

import androidx.compose.ui.unit.dp

import androidx.compose.ui.unit.sp

import com.example.happybirthday.ui.theme.HappyBirthdayTheme

// Clase principal que representa la pantalla de la app

class MainActivity : ComponentActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

// Define el contenido de la actividad usando Jetpack Compose

setContent {

// Aplica el tema personalizado definido para la app

HappyBirthdayTheme {

// Superficie que ocupa toda la pantalla con el color de fondo del tema

Surface(

modifier = Modifier.fillMaxSize(),

color = MaterialTheme.colorScheme.background

) {

// Llama al composable que muestra los textos de felicitación

GreetingText(

message = "Happy Birthday Sam!", // Mensaje principal

from = "From Emma", // Firma del mensaje

modifier = Modifier.padding(8.dp) // Añade padding alrededor

)

}

}

}

}

}

// Composable que muestra el mensaje de cumpleaños y la firma

@Composable

fun GreetingText(message: String, from: String, modifier: Modifier = Modifier) {

// Columna vertical para organizar los textos uno debajo del otro

Column(

verticalArrangement = Arrangement.Center, // Centra el contenido verticalmente

modifier = modifier

) {

// Texto principal del mensaje de cumpleaños

Text(

text = message,

fontSize = 100.sp, // Tamaño de fuente muy grande

lineHeight = 116.sp, // Altura de línea para mejor lectura

textAlign = TextAlign.Center // Centra el texto horizontalmente

)

// Texto con el nombre de quien envía la felicitación

Text(

text = from,

fontSize = 36.sp, // Tamaño de fuente para la firma

modifier = Modifier

.padding(16.dp) // Padding para separar del texto anterior

.align(Alignment.End) // Alinea la firma al final (derecha)

)

}

}

// Vista previa en el editor de Android Studio sin necesidad de ejecutar en emulador

@Preview(showBackground = true)

@Composable

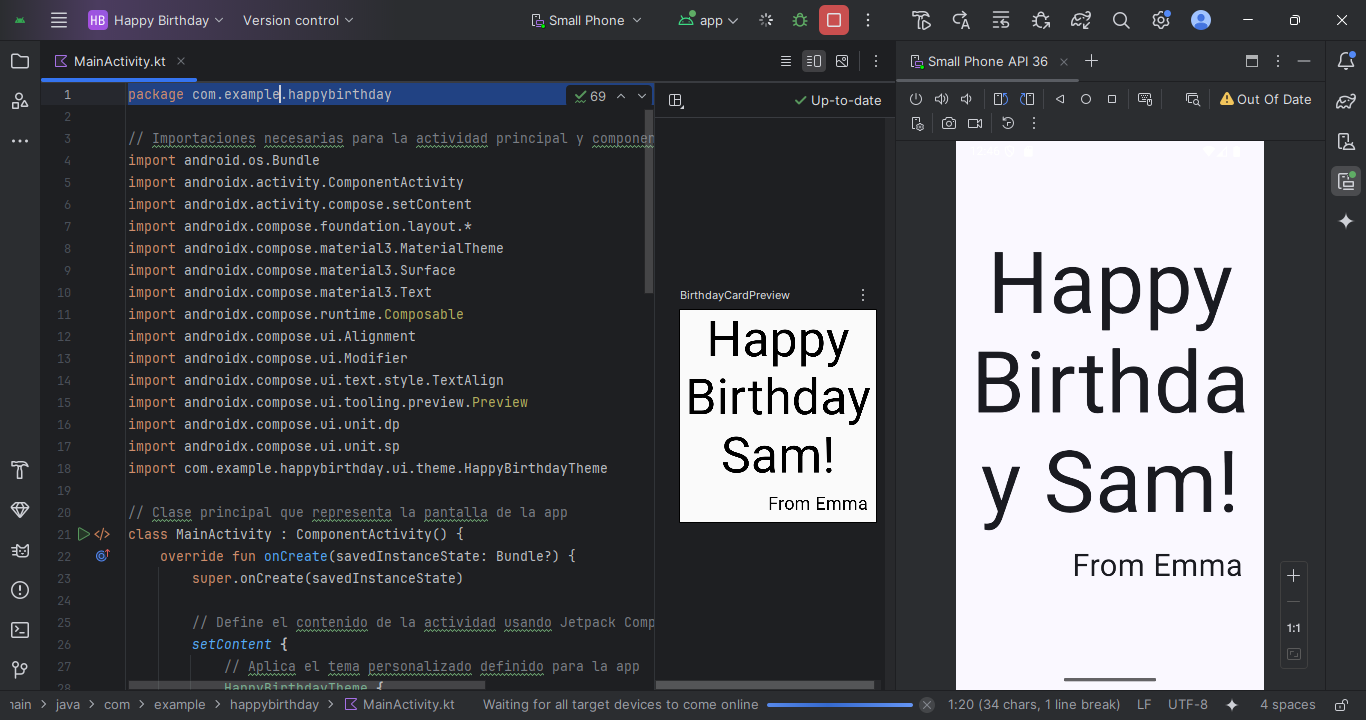
fun BirthdayCardPreview() {

HappyBirthdayTheme {

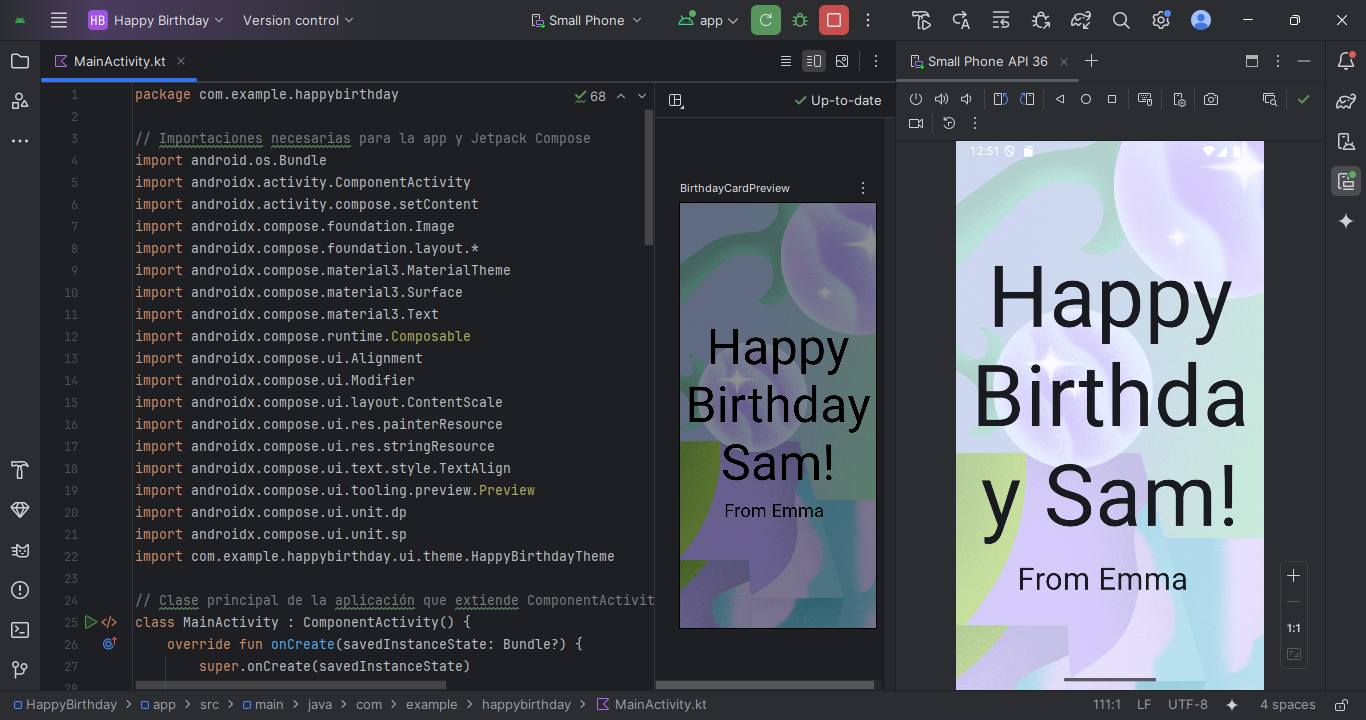
// Muestra una tarjeta de ejemplo con valores de prueba

GreetingText(message = "Happy Birthday Sam!", from = "From Emma")

}

} 

package com.example.happybirthday  
// Importaciones necesarias para la app y Jetpack Compose  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.compose.foundation.Image  
import androidx.compose.foundation.layout.\*  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.Surface  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.layout.ContentScale  
import androidx.compose.ui.res.painterResource  
import androidx.compose.ui.res.stringResource  
import androidx.compose.ui.text.style.TextAlign  
import androidx.compose.ui.tooling.preview.Preview  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import com.example.happybirthday.ui.theme.HappyBirthdayTheme  
// Clase principal de la aplicación que extiende ComponentActivity  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 // Establece el contenido de la pantalla con Jetpack Compose  
 *setContent* **{** // Se aplica el tema personalizado de la app  
 HappyBirthdayTheme **{** // Superficie que ocupa toda la pantalla con el color de fondo del tema  
 Surface(  
 modifier = Modifier.*fillMaxSize*(),  
 color = MaterialTheme.colorScheme.background  
 ) **{** // Llama al composable que muestra la imagen y el texto  
 GreetingImage(  
 stringResource(R.string.*happy\_birthday\_text*), // Mensaje principal  
 stringResource(R.string.*signature\_text*) // Firma o autor del mensaje  
 )  
 **}  
 }  
 }** }  
}  
  
// Composable que muestra los textos del saludo  
@Composable  
fun GreetingText(message: String, from: String, modifier: Modifier = Modifier) {  
 // Columna vertical para evitar que los textos se superpongan  
 Column(  
 verticalArrangement = Arrangement.Center,  
 modifier = modifier  
 ) **{** // Texto principal del saludo (ej. "Happy Birthday!")  
 Text(  
 text = message,  
 fontSize = 100.*sp*,  
 lineHeight = 116.*sp*,  
 textAlign = TextAlign.Center,  
 modifier = Modifier.*padding*(top = 16.*dp*)  
 )  
 // Texto de firma (ej. "From Sergio")  
 Text(  
 text = from,  
 fontSize = 36.*sp*,  
 modifier = Modifier  
 .*padding*(top = 16.*dp*)  
 .*padding*(end = 16.*dp*)  
 .*align*(Alignment.CenterHorizontally) // Centra horizontalmente dentro de la columna  
 )  
 **}**}  
  
// Composable que combina la imagen de fondo con los textos del saludo  
@Composable  
fun GreetingImage(message: String, from: String, modifier: Modifier = Modifier) {  
 // Caja que permite superponer elementos (imagen + texto)  
 Box(modifier) **{** // Imagen de fondo (por ejemplo, fiesta de Android)  
 Image(  
 painter = painterResource(id = R.drawable.*androidparty*), // Imagen en drawable  
 contentDescription = null, // No necesita descripción accesible en este caso  
 contentScale = ContentScale.Crop, // Recorta la imagen para llenar el contenedor  
 alpha = 0.5F // Opacidad reducida para que el texto sea más visible  
 )  
 // Llamada al composable de texto  
 GreetingText(  
 message = message,  
 from = from,  
 modifier = Modifier  
 .*fillMaxSize*()  
 .*padding*(8.*dp*)  
 )  
 **}**}  
  
// Vista previa para el editor sin necesidad de ejecutar la app en el emulador  
@Preview(showBackground = false)  
@Composable  
private fun BirthdayCardPreview() {  
 HappyBirthdayTheme **{** GreetingImage(  
 stringResource(R.string.*happy\_birthday\_text*),  
 stringResource(R.string.*signature\_text*)  
 )  
 **}**}



package com.example.jetpackcompose

// Importaciones necesarias para una actividad con Jetpack Compose

import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.Image

import androidx.compose.foundation.layout.\*

import androidx.compose.material3.MaterialTheme

import androidx.compose.material3.Surface

import androidx.compose.material3.Text

import androidx.compose.runtime.Composable

import androidx.compose.ui.Modifier

import androidx.compose.ui.layout.ContentScale

import androidx.compose.ui.res.painterResource

import androidx.compose.ui.res.stringResource

import androidx.compose.ui.text.style.TextAlign

import androidx.compose.ui.tooling.preview.Preview

import androidx.compose.ui.unit.dp

import androidx.compose.ui.unit.sp

import com.example.jetpackcompose.ui.theme.JetpackComposeTheme

// Actividad principal de la aplicación

class MainActivity : ComponentActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

// Define el contenido de la actividad utilizando Jetpack Compose

setContent {

// Se aplica el tema personalizado

JetpackComposeTheme {

// Crea una superficie que ocupa toda la pantalla

Surface(

modifier = Modifier.fillMaxSize(),

color = MaterialTheme.colorScheme.background // Usa el color de fondo del tema

) {

// Llama a la función composable que muestra el artículo

ArticleScreen()

}

}

}

}

}

// Función composable que define la estructura del artículo (imagen + texto)

@Composable

fun ArticleScreen(modifier: Modifier = Modifier) {

// Organiza los elementos en una columna (de arriba hacia abajo)

Column(modifier = modifier) {

// Muestra una imagen encabezado

Image(

painter = painterResource(id = R.drawable.compose\_header), // Imagen desde recursos

contentDescription = null, // No se necesita descripción (decorativa)

contentScale = ContentScale.FillWidth, // Escala la imagen al ancho del contenedor

modifier = Modifier.fillMaxWidth()

)

// Título del artículo

Text(

text = stringResource(R.string.title), // Texto desde recursos strings.xml

fontSize = 24.sp, // Tamaño de fuente

modifier = Modifier.padding(16.dp) // Margen alrededor del texto

)

// Primer párrafo del artículo

Text(

text = stringResource(R.string.paragraph1),

textAlign = TextAlign.Justify, // Justifica el texto

modifier = Modifier.padding(horizontal = 16.dp, vertical = 8.dp) // Padding horizontal y vertical

)

// Segundo párrafo del artículo

Text(

text = stringResource(R.string.paragraph2),

textAlign = TextAlign.Justify,

modifier = Modifier.padding(16.dp)

)

}

}

// Vista previa del diseño sin ejecutar la app

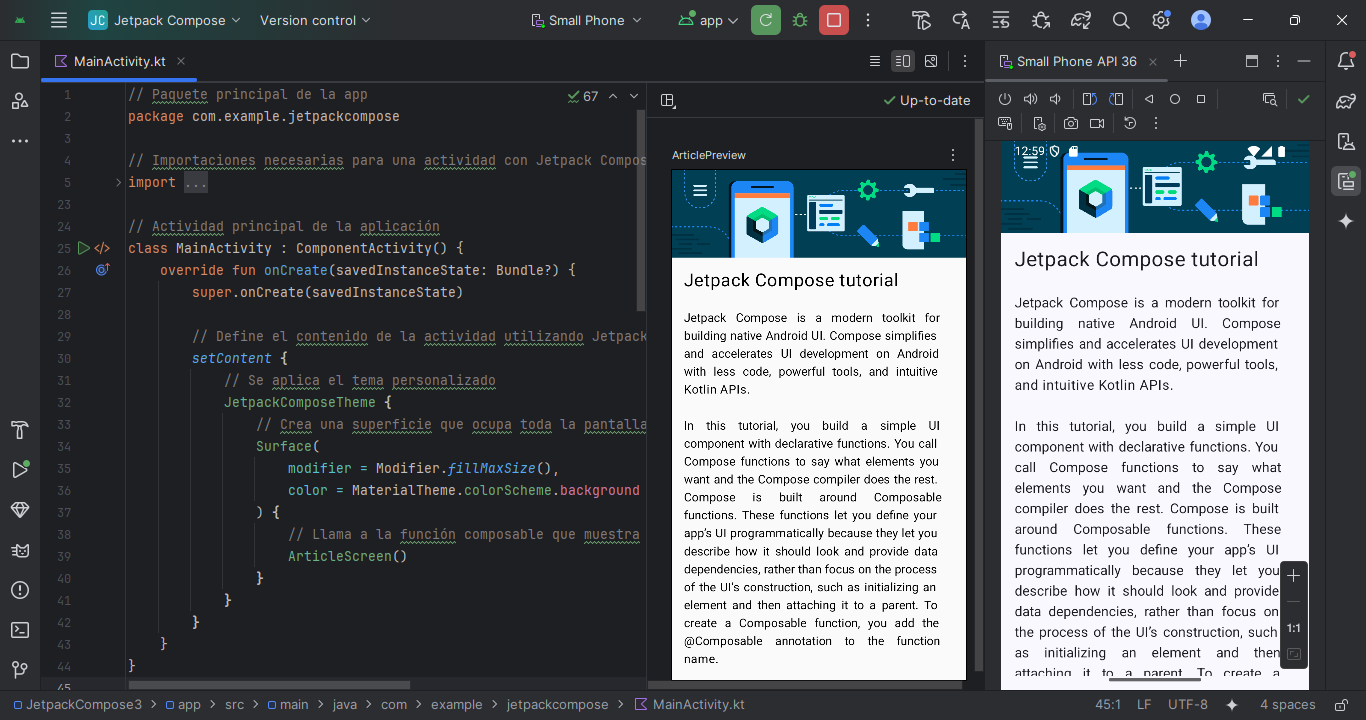
@Preview(showBackground = true)

@Composable

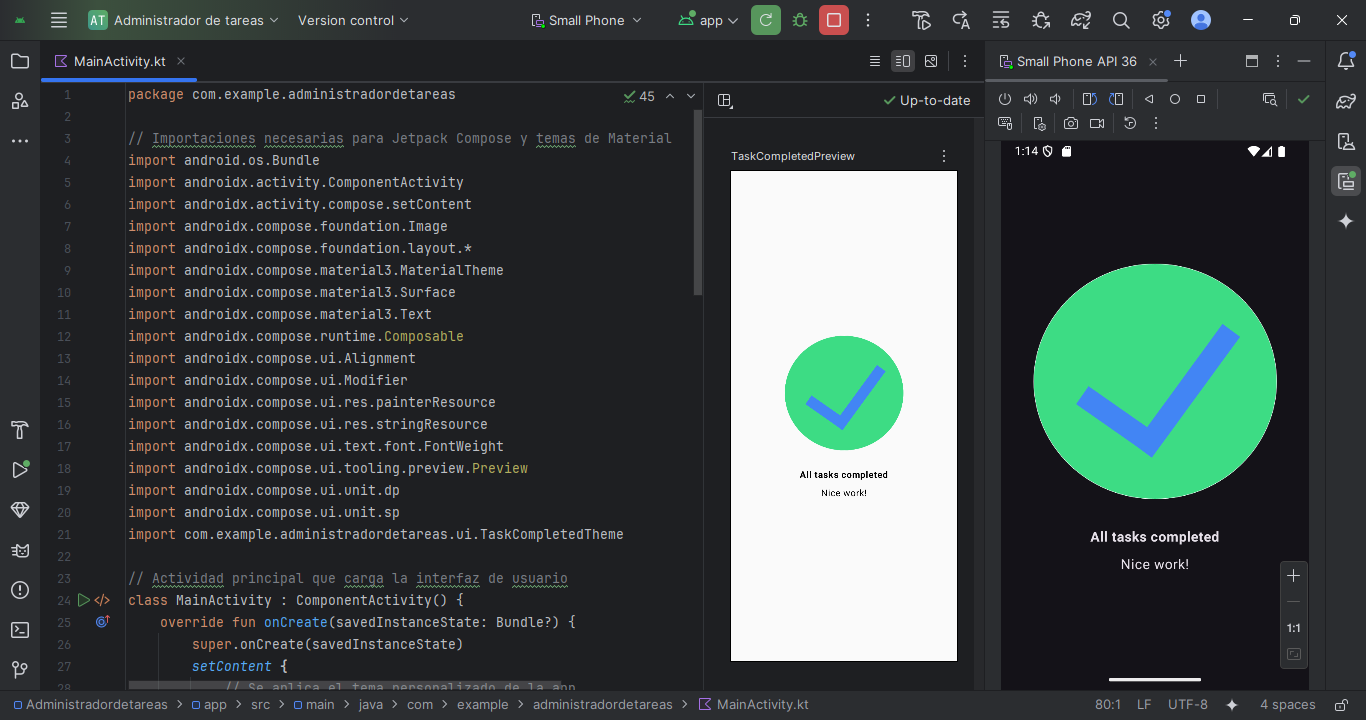
fun ArticlePreview() {

JetpackComposeTheme {

ArticleScreen() // Muestra la vista previa del artículo

 }

}

package com.example.administradordetareas  
// Importaciones necesarias para Jetpack Compose y temas de Material  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.compose.foundation.Image  
import androidx.compose.foundation.layout.\*  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.Surface  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.res.painterResource  
import androidx.compose.ui.res.stringResource  
import androidx.compose.ui.text.font.FontWeight  
import androidx.compose.ui.tooling.preview.Preview  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import com.example.administradordetareas.ui.TaskCompletedTheme  
// Actividad principal que carga la interfaz de usuario  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *setContent* **{** // Se aplica el tema personalizado de la app  
 TaskCompletedTheme **{** // Superficie que ocupa toda la pantalla con el color de fondo del tema  
 Surface(  
 modifier = Modifier.*fillMaxSize*(),  
 color = MaterialTheme.colorScheme.background  
 ) **{** // Llama al componente composable principal  
 TaskCompletedScreen()  
 **}  
 }  
 }** }  
}  
  
// Función composable que muestra la pantalla de tareas completadas  
@Composable  
fun TaskCompletedScreen() {  
 Column(  
 modifier = Modifier  
 .*fillMaxSize*(), // Llena toda la pantalla  
 verticalArrangement = Arrangement.Center, // Centrado vertical  
 horizontalAlignment = Alignment.CenterHorizontally // Centrado horizontal  
 ) **{** // Imagen representando que la tarea está completada  
 Image(  
 painter = painterResource(R.drawable.*ic\_task\_completed*),  
 contentDescription = null // No se requiere descripción si es decorativa  
 )  
 // Primer texto con estilo en negrita y padding  
 Text(  
 text = stringResource(R.string.*all\_task\_completed*),  
 modifier = Modifier.*padding*(top = 24.*dp*, bottom = 8.*dp*),  
 fontWeight = FontWeight.Bold  
 )  
 // Segundo texto con tamaño de fuente personalizado  
 Text(  
 text = stringResource(R.string.*nice\_work*),  
 fontSize = 16.*sp* )  
 **}**}  
// Vista previa de la pantalla para el modo diseño  
@Preview(showBackground = true)  
@Composable  
fun TaskCompletedPreview() {  
 TaskCompletedTheme **{** TaskCompletedScreen()  
 **}**}

// Paquete principal del proyecto

package com.example.cuadrado

// Importaciones necesarias para Jetpack Compose y Material3

import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.background

import androidx.compose.foundation.layout.\*

import androidx.compose.material3.MaterialTheme

import androidx.compose.material3.Surface

import androidx.compose.material3.Text

import androidx.compose.runtime.Composable

import androidx.compose.ui.Alignment

import androidx.compose.ui.Modifier

import androidx.compose.ui.graphics.Color

import androidx.compose.ui.text.font.FontWeight

import androidx.compose.ui.unit.dp

import androidx.compose.ui.unit.sp

import com.example.cuadrado.ui.theme.CuadradoTheme

// Actividad principal del proyecto

class MainActivity : ComponentActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

// Establece el contenido de la pantalla usando Jetpack Compose

setContent {

CuadradoTheme { // Aplica el tema definido en ui.theme

Surface(

modifier = Modifier.fillMaxSize(),

color = MaterialTheme.colorScheme.background // Fondo del tema

) {

CuadranteCompose() // Llama a la función principal de la UI

}

}

}

}

}

// Composable que construye la pantalla dividida en 4 cuadrantes

@Composable

fun CuadranteCompose() {

Column(Modifier.fillMaxSize()) { // Columna que ocupa toda la pantalla

// Primera fila con dos columnas (cuadrantes superiores)

Row(Modifier.weight(1f)) {

Cuadro(

title = "Texto Composable", // Título del primer cuadro

description = "Muestra texto en la pantalla", // Descripción

backgroundColor = Color(0xFFEADDFF), // Color de fondo

modifier = Modifier.weight(1f) // Ocupa la mitad del ancho

)

Cuadro(

title = "Image Composable",

description = "Muestra una imagen",

backgroundColor = Color(0xFFD0BCFF),

modifier = Modifier.weight(1f)

)

}

// Segunda fila con dos columnas (cuadrantes inferiores)

Row(Modifier.weight(1f)) {

Cuadro(

title = "Row Composable",

description = "Organiza los elementos en una fila horizontal",

backgroundColor = Color(0xFFB69DF8),

modifier = Modifier.weight(1f)

)

Cuadro(

title = "Column Composable",

description = "Organiza los elementos en una columna vertical",

backgroundColor = Color(0xFFF6EDFF),

modifier = Modifier.weight(1f)

)

}

}

}

// Composable reutilizable para cada cuadrante

@Composable

fun Cuadro(

title: String, // Texto en negrita

description: String, // Texto descriptivo

backgroundColor: Color, // Color de fondo del cuadro

modifier: Modifier = Modifier // Modificador externo para tamaño y posición

) {

Column(

modifier // Se aplica al contenedor completo

.fillMaxSize() // Ocupa todo el espacio disponible del cuadrante

.background(color = backgroundColor) // Aplica color de fondo

.padding(16.dp), // Espaciado interno

verticalArrangement = Arrangement.Center, // Centrado vertical

horizontalAlignment = Alignment.CenterHorizontally // Centrado horizontal

) {

// Título en negrita

Text(

text = title,

fontWeight = FontWeight.Bold,

fontSize = 18.sp,

modifier = Modifier.padding(bottom = 8.dp) // Espacio inferior

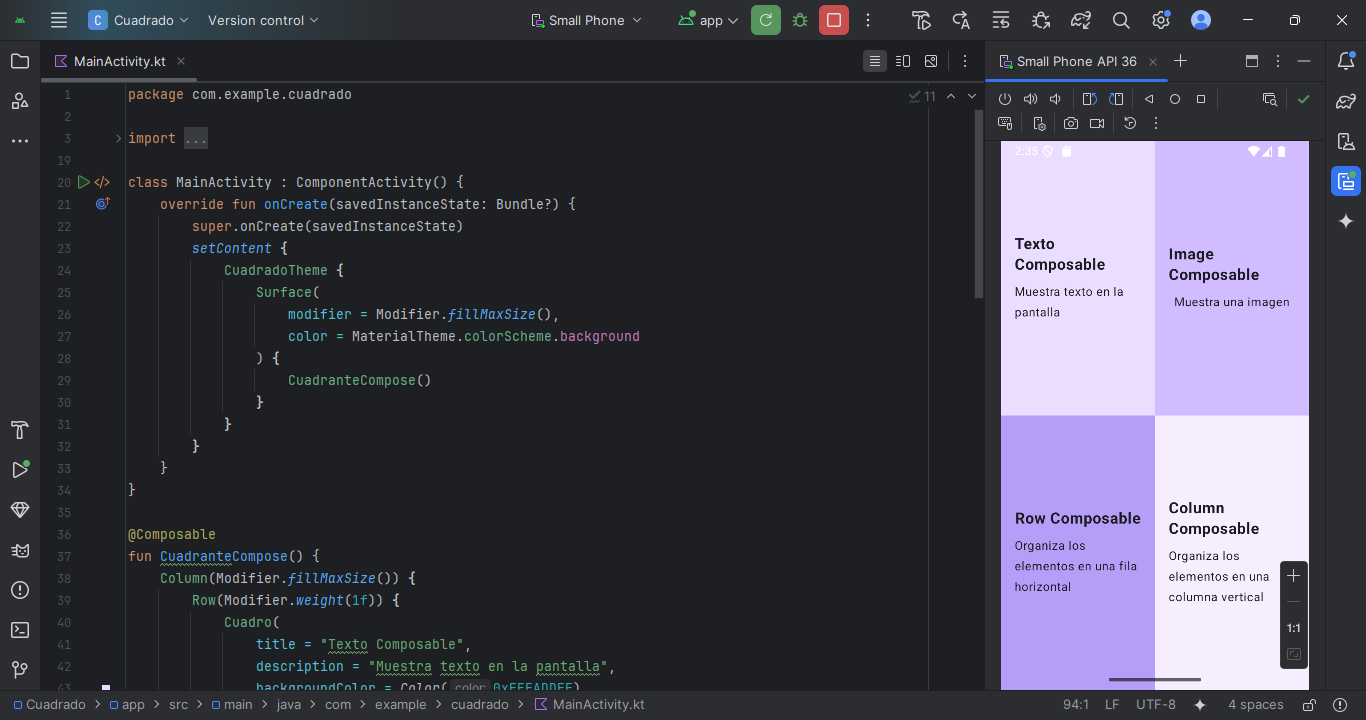
)

// Texto descriptivo

Text(text = description, fontSize = 14.sp)

}

}



package com.example.tarjetasdepresentacin  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.compose.foundation.Image  
import androidx.compose.foundation.layout.\*  
import androidx.compose.material.icons.Icons  
import androidx.compose.material.icons.filled.*Email*import androidx.compose.material.icons.filled.*Phone*import androidx.compose.material.icons.filled.*Share*import androidx.compose.material3.\*  
import androidx.compose.runtime.Composable  
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.unit.dp  
import androidx.compose.ui.unit.sp  
import androidx.compose.ui.tooling.preview.Preview  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 // Aquí se define el contenido de la UI utilizando Jetpack Compose  
 *setContent* **{** BusinessCardApp() // Llama a la función composable principal que define la tarjeta  
 **}** }  
}  
// Función composable principal que estructura toda la interfaz de la tarjeta de presentación  
@Composable  
fun BusinessCardApp() {  
 // Surface actúa como el contenedor de fondo para toda la pantalla  
 Surface(  
 modifier = Modifier.*fillMaxSize*(), // Ocupa toda la pantalla  
 color = *Color*(0xFF073042) // Color de fondo oscuro  
 ) **{** // Column organiza los elementos de forma vertical  
 Column(  
 verticalArrangement = Arrangement.Center, // Centra verticalmente el contenido  
 horizontalAlignment = Alignment.CenterHorizontally, // Centra horizontalmente  
 modifier = Modifier.*fillMaxSize*()  
 ) **{** ProfileSection() // Muestra la sección del perfil (nombre, título, imagen)  
 Spacer(modifier = Modifier.*height*(32.*dp*)) // Espacio entre secciones  
 ContactSection() // Muestra la sección de contacto con íconos y texto  
 **}  
 }**}  
// Función composable que define la sección del perfil (imagen, nombre, título)  
@Composable  
fun ProfileSection() {  
 Column(  
 horizontalAlignment = Alignment.CenterHorizontally // Centra el contenido horizontalmente  
 ) **{** // Imagen del logo de Android (recurso localizado en drawable)  
 Image(  
 painter = painterResource(id = R.drawable.*android\_logo*),

contentDescription = "Logo de Android",  
 modifier = Modifier  
 .*size*(100.*dp*) // Tamaño de la imagen en dp (density-independent pixels)  
 )  
 // Nombre completo del usuario en texto grande y negrita  
 Text(  
 text = "Sergio Daniel",  
 fontSize = 36.*sp*,  
 color = Color.White,  
 fontWeight = FontWeight.Bold  
 )  
 // Título o profesión debajo del nombre, con color verde estilo Android  
 Text(  
 text = "Ing. Sistemas Computacionales",  
 fontSize = 20.*sp*,  
 color = *Color*(0xFF3ddc84) // Verde Android  
 )  
 **}**}  
// Función composable que muestra la sección de contacto (teléfono, red social, correo)  
@Composable  
fun ContactSection() {  
 Column(  
 modifier = Modifier.*padding*(horizontal = 16.*dp*) // Margen horizontal  
 ) **{** // Fila para el teléfono  
 ContactInfoRow(icon = Icons.Default.*Phone*, contactText = "+52 833 181 4545”  
 // Fila para red social o nombre de usuario  
 ContactInfoRow(icon = Icons.Default.*Share*, contactText = "@Luis Fernando Hernández Mercado")  
 // Fila para dirección de correo electrónico  
 ContactInfoRow(icon = Icons.Default.*Email*, contactText = "L20070531@cdmadero.tecnm.mx")  
 **}**}  
// Función reusable que representa una fila de información de contacto con ícono y texto  
@Composable  
fun ContactInfoRow(icon: androidx.compose.ui.graphics.vector.ImageVector, contactText: String) {  
 Row(  
 verticalAlignment = Alignment.CenterVertically, // Alineación vertical centrada  
 modifier = Modifier  
 .*padding*(vertical = 8.*dp*) // Espaciado entre filas  
 ) **{** // Ícono representativo del tipo de contacto (teléfono, email, red social)  
 Icon(  
 imageVector = icon,  
 contentDescription = null, // No es necesario para accesibilidad en este contexto  
 tint = *Color*(0xFF3ddc84), // Color verde Android para consistencia visual  
 modifier = Modifier.*size*(24.*dp*)  
 )  
 Spacer(modifier = Modifier.*width*(16.*dp*)) // Espacio entre el ícono y el texto  
 // Texto que muestra la información de contacto  
 Text(  
 text = contactText,  
 color = Color.White,  
 fontSize = 16.*sp* )  
 **}**}  
// Función para previsualizar el diseño en Android Studio sin necesidad de ejecutarlo en un dispositivo  
@Preview(showBackground = true)  
@Composable  
fun BusinessCardPreview() {  
 BusinessCardApp()  
}

