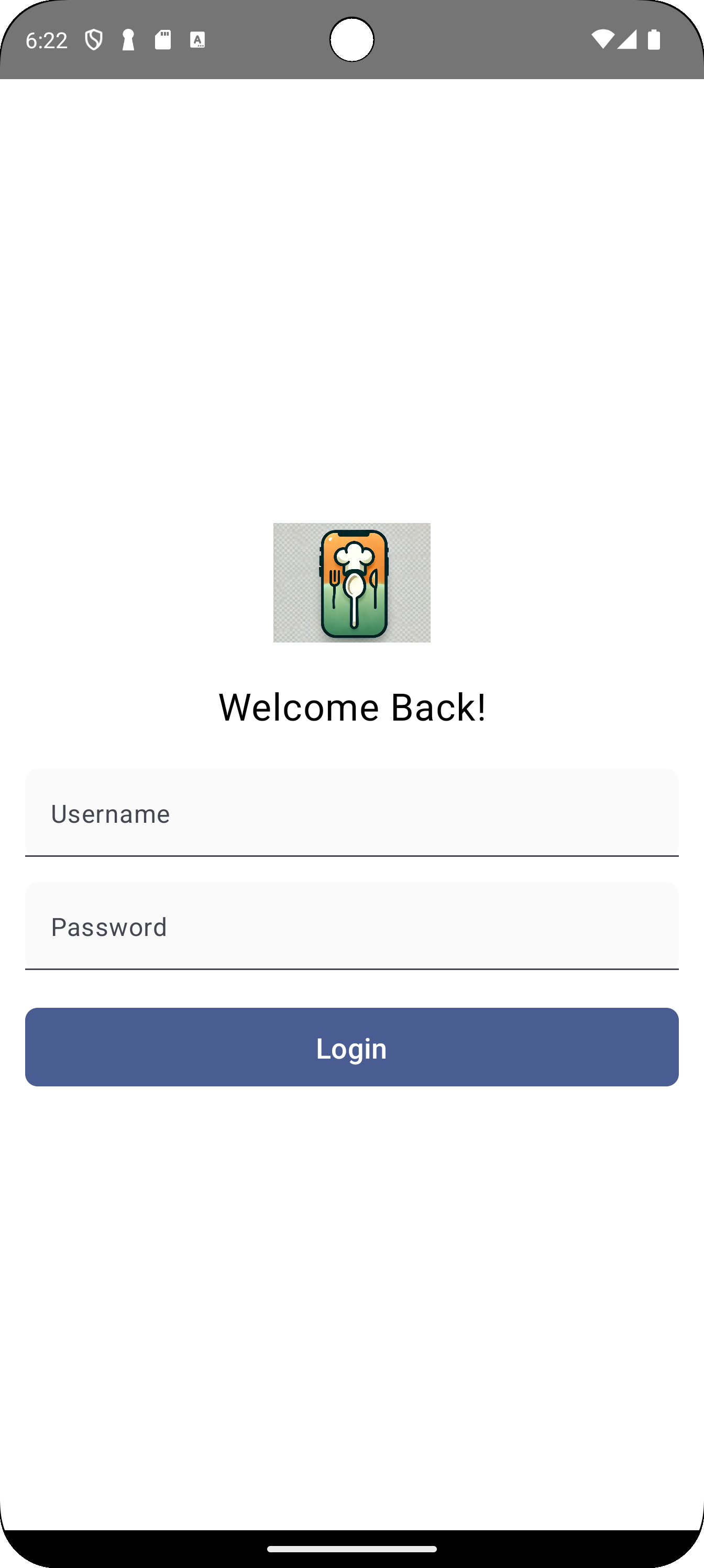
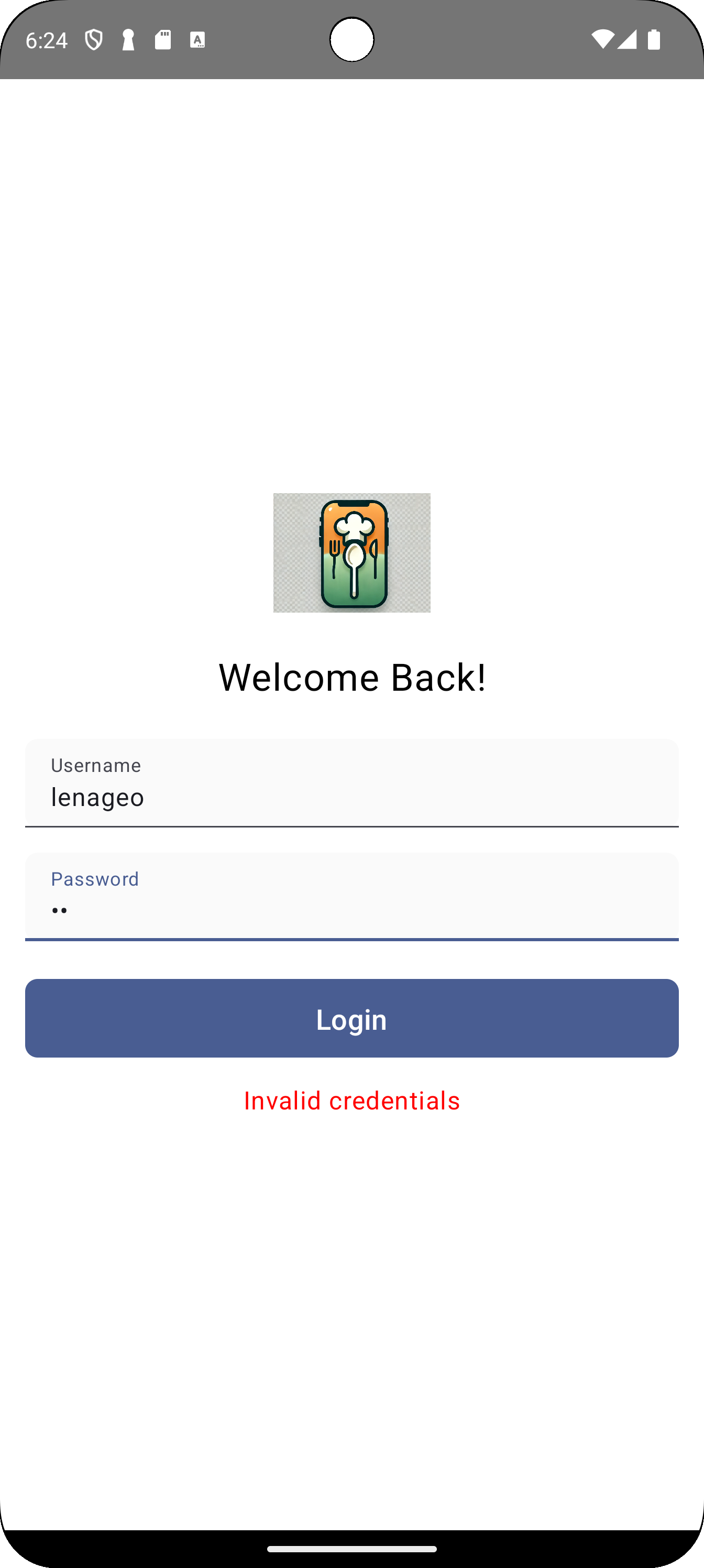
**MOBILE APPLICATIONS LAB 6- CREATING UI WITH COMPOSABLE VIEWS**

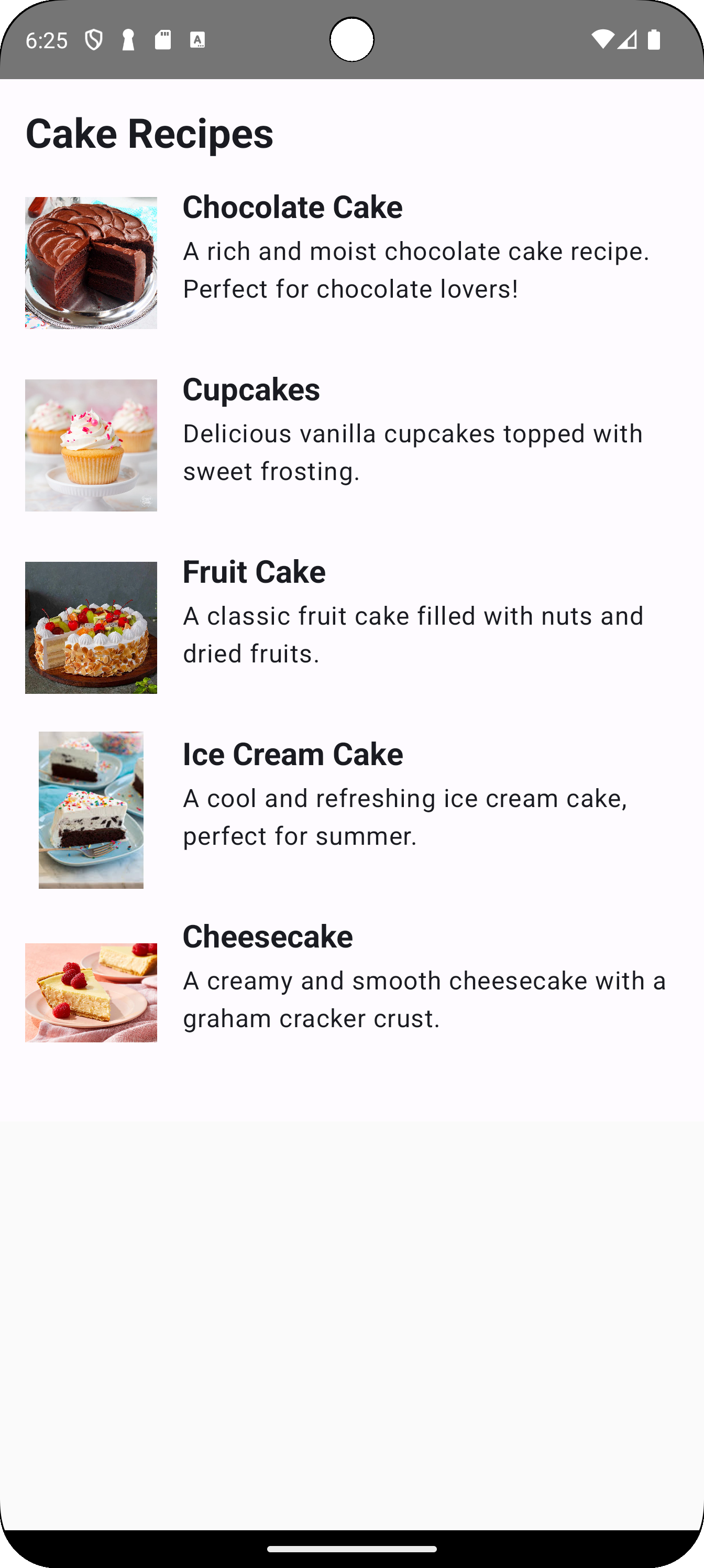
**NAME: LENA GEO**

**REGISTER NUMBER: 2241139**

**OUTPUT:**

****

****

****

**CODE:**

**MainActivity.kt**

package com.example.lab6  
  
import android.content.Intent  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.compose.foundation.Image  
import androidx.compose.foundation.background  
import androidx.compose.foundation.layout.\*  
import androidx.compose.foundation.shape.RoundedCornerShape  
import androidx.compose.material3.\*  
import androidx.compose.runtime.\*  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.graphics.Color  
import androidx.compose.ui.layout.ContentScale  
import androidx.compose.ui.platform.*LocalContext*import androidx.compose.ui.res.painterResource  
import androidx.compose.ui.text.input.PasswordVisualTransformation  
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.lab6.ui.theme.Lab6Theme  
  
@OptIn(ExperimentalMaterial3Api::class) // <- Add this annotation  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *setContent* {  
 Lab6Theme {  
 LoginPage()  
 }  
 }  
 }  
}  
  
@OptIn(ExperimentalMaterial3Api::class) // <- Add this annotation to the composable  
@Composable  
fun LoginPage() {  
 var username by remember { *mutableStateOf*("") }  
 var password by remember { *mutableStateOf*("") }  
 var errorMessage by remember { *mutableStateOf*("") }  
  
 val context = *LocalContext*.current  
  
 Box(  
 modifier = Modifier  
 .*fillMaxSize*()  
 .*background*(Color.White)  
 .*padding*(16.*dp*),  
 contentAlignment = Alignment.Center  
 ) {  
 Column(  
 horizontalAlignment = Alignment.CenterHorizontally,  
 verticalArrangement = Arrangement.Center,  
 ) {  
 // Logo  
 Image(  
 painter = painterResource(id = R.drawable.*logo*),  
 contentDescription = "App Logo",  
 modifier = Modifier  
 .*size*(100.*dp*)  
 .*padding*(bottom = 24.*dp*),  
 contentScale = ContentScale.Crop  
 )  
  
 // Title  
 Text(  
 text = "Welcome Back!",  
 fontSize = 24.*sp*,  
 color = Color.Black,  
 textAlign = TextAlign.Center,  
 modifier = Modifier.*padding*(bottom = 24.*dp*)  
 )  
  
 // Username Field  
 TextField(  
 value = username,  
 onValueChange = { username = it },  
 label = { Text("Username") },  
 singleLine = true,  
 shape = *RoundedCornerShape*(8.*dp*),  
 colors = TextFieldDefaults.textFieldColors(  
 containerColor = Color.LightGray.copy(alpha = 0.1f)  
 ),  
 modifier = Modifier  
 .*fillMaxWidth*()  
 )  
  
 Spacer(modifier = Modifier.*height*(16.*dp*))  
  
 // Password Field  
 TextField(  
 value = password,  
 onValueChange = { password = it },  
 label = { Text("Password") },  
 singleLine = true,  
 shape = *RoundedCornerShape*(8.*dp*),  
 visualTransformation = PasswordVisualTransformation(),  
 colors = TextFieldDefaults.textFieldColors(  
 containerColor = Color.LightGray.copy(alpha = 0.1f)  
 ),  
 modifier = Modifier  
 .*fillMaxWidth*()  
 )  
  
 Spacer(modifier = Modifier.*height*(24.*dp*))  
  
 // Login Button  
 Button(  
 onClick = {  
 if (username.*isEmpty*() || password.*isEmpty*()) {  
 errorMessage = "Please fill in all fields"  
 } else {  
 if (username == "lenageo" && password == "test") {  
 // Use context to start the new activity  
 val intent = Intent(context, RecipeActivity::class.*java*)  
 context.startActivity(intent)  
 } else {  
 errorMessage = "Invalid credentials"  
 }  
 }  
 },  
 shape = *RoundedCornerShape*(8.*dp*),  
 modifier = Modifier  
 .*fillMaxWidth*()  
 .*height*(50.*dp*)  
 ) {  
 Text(text = "Login", fontSize = 18.*sp*)  
 }  
  
 // Error Message  
 if (errorMessage.*isNotEmpty*()) {  
 Spacer(modifier = Modifier.*height*(16.*dp*))  
 Text(text = errorMessage, color = Color.Red)  
 }  
 }  
 }  
}  
  
@Preview(showBackground = true)  
@Composable  
fun PreviewLoginPage() {  
 Lab6Theme {  
 LoginPage()  
 }  
}

**RecipeActivity.kt**

package com.example.lab6  
  
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.foundation.rememberScrollState  
import androidx.compose.foundation.verticalScroll  
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.res.painterResource  
import androidx.compose.ui.text.font.FontWeight  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import com.example.lab6.ui.theme.Lab6Theme  
  
class RecipeActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *setContent* {  
 Lab6Theme {  
 Surface(color = MaterialTheme.colorScheme.background) {  
 RecipeContent()  
 }  
 }  
 }  
 }  
  
 @Composable  
 fun RecipeContent() {  
 Column(  
 modifier = Modifier  
 .*padding*(16.*dp*)  
 .*verticalScroll*(rememberScrollState())  
 ) {  
 // Title for the recipe list  
 Text(  
 text = "Cake Recipes",  
 style = MaterialTheme.typography.headlineMedium.copy(fontWeight = FontWeight.Bold),  
 fontSize = 26.*sp*,  
 modifier = Modifier.*padding*(bottom = 16.*dp*)  
 )  
  
 // Recipe items  
 RecipeItem(  
 imageRes = R.drawable.*chocolate\_cake*,  
 title = "Chocolate Cake",  
 description = "A rich and moist chocolate cake recipe. Perfect for chocolate lovers!"  
 )  
  
 RecipeItem(  
 imageRes = R.drawable.*cupcake*,  
 title = "Cupcakes",  
 description = "Delicious vanilla cupcakes topped with sweet frosting."  
 )  
  
 RecipeItem(  
 imageRes = R.drawable.*fruit\_cake*,  
 title = "Fruit Cake",  
 description = "A classic fruit cake filled with nuts and dried fruits."  
 )  
  
 // Add more recipes here  
 RecipeItem(  
 imageRes = R.drawable.*ice\_cream\_cake*,  
 title = "Ice Cream Cake",  
 description = "A cool and refreshing ice cream cake, perfect for summer."  
 )  
  
 RecipeItem(  
 imageRes = R.drawable.*cheescake*,  
 title = "Cheesecake",  
 description = "A creamy and smooth cheesecake with a graham cracker crust."  
 )  
 }  
 }  
  
 @Composable  
 fun RecipeItem(imageRes: Int, title: String, description: String) {  
 Row(modifier = Modifier.*padding*(bottom = 16.*dp*)) {  
 Image(  
 painter = painterResource(id = imageRes),  
 contentDescription = title,  
 modifier = Modifier  
 .*size*(100.*dp*)  
 .*padding*(end = 16.*dp*)  
 )  
  
 Column(  
 modifier = Modifier.*fillMaxWidth*()  
 ) {  
 Text(  
 text = title,  
 style = MaterialTheme.typography.headlineMedium.copy(fontWeight = FontWeight.Bold),  
 fontSize = 20.*sp* )  
 Spacer(modifier = Modifier.*height*(4.*dp*))  
 Text(  
 text = description,  
 style = MaterialTheme.typography.bodyLarge.copy(fontSize = 16.*sp*)  
 )  
 }  
 }  
 }  
}