```kotlin
package com.example.login

import androidx.compose.foundation.BorderStroke
import androidx.compose.foundation.Image
import androidx.compose.foundation.layout.*
import androidx.compose.material.*
import androidx.compose.runtime.Composable
import androidx.compose.runtime.mutableStateOf
import androidx.compose.runtime.remember
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.TextStyle
import androidx.compose.ui.text.input.TextFieldValue
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp

@Composable
fun LoginScreen(name: String) {
    var islogin = remember {
        mutableStateOf(false)
    }
    Scaffold(content = {
        Column() {
            Column(modifier= Modifier.padding(20.dp),
                verticalArrangement = Arrangement.Center ,
                horizontalAlignment = Alignment.CenterHorizontally)
{

                Text(text = "English (US)" , style =
TextStyle(fontSize = 10.sp))
                Spacer(modifier = Modifier.height(10.dp))
                Image(painter = painterResource(id =
R.drawable.fblogo), contentDescription = null,
                    modifier = Modifier.size(100.dp))

                Spacer(modifier = Modifier.height(20.dp))
                val textState = remember {
                    mutableStateOf(TextFieldValue())
                }

                OutlinedTextField(
                    value = textState.value, onValueChange = {
textState.value = it },
                    modifier = Modifier.fillMaxWidth(),
```

```kotlin
                    placeholder = { Text(text = "Mobile number or
email") },
                    colors =
TextFieldDefaults.outlinedTextFieldColors(
                        focusedBorderColor = Color.White
                    )
                )

                Spacer(modifier = Modifier.height(10.dp))

                OutlinedTextField(
                    value = textState.value, onValueChange = {
textState.value = it },
                    modifier = Modifier.fillMaxWidth(),
                    placeholder = { Text(text = "Password") },
                    colors =
TextFieldDefaults.outlinedTextFieldColors(
                        focusedBorderColor = Color.White)
                )

                Spacer(modifier = Modifier.height(10.dp))

                Button(
                    onClick = { islogin.value = !islogin.value
                        //  navController.navigate(route =
Route.HomeScreen +"/Something")
//                    HomeScreen(name = "dfkjskalga")
                    },
                    colors= ButtonDefaults.textButtonColors(
backgroundColor= Color(59,89,152)),
                    //shape = RoundedCornerShape(50.dp),
                    modifier = Modifier
                        .fillMaxWidth()
                        .height(50.dp),
                ) {
                    //if(islogin.value){
                    // CircularProgressIndicator()

                    // }else{
                    Text(text = "Login", color= Color(255,255,255))
                    // }

                }

                TextButton(onClick = { /*TODO*/ }) {
                    Text(text = "Find your account or password",
```

```kotlin
                        color= Color(255,255,255)
                    )
                }
            }

            Column(modifier= Modifier.padding(20.dp),
                verticalArrangement = Arrangement.Bottom,
                horizontalAlignment = Alignment.CenterHorizontally
            ){

                OutlinedButton(onClick = { /*TODO*/ },
                    modifier = Modifier
                        .fillMaxWidth()
                        .height(50.dp),
                    border = BorderStroke(2.dp, Color(59,89,152))
                ) {
                    Text(text = "Create new account",
                        color= Color(255,255,255)
                    )
                }
                Spacer(modifier = Modifier.height(10.dp))


                Row() {
                    Image(painter = painterResource(id =
R.drawable.metalogo), contentDescription = null,
                        modifier = Modifier.size(20.dp))
                    Text(text = " Meta")

                }
            }

        }
    })
    if(islogin.value){
        HomeScreen(name = "iooppp")
    }
}
```