



PSG INSTITUTE OF TECHNOLOGY AND APPLIED RESEARCH  
COIMBATORE-641062

B.Tech - Artificial Intelligence and Data Science

# Snack Squad: A Customizable Snack Ordering and Delivery App

Submitted by :

Kavitha S(715522243024)

PrathibhaM(715522243036)

Suvathi T(715522243054)

Kishore J(715522243302)

RatnavelM(715522243304)



## Description:

This project involves developing Snack Squad, a sample e-commerce application using Android Jetpack Compose, a modern toolkit for building native Android UIs. The app provides a simple and interactive experience where users can register, log in, browse a list of snacks, add items to their cart, and proceed to checkout. Additionally, an admin login allows order management and tracking. The project demonstrates key aspects of app development, including user authentication, database integration for storing user and order data, and building dynamic, visually appealing UIs with Compose. By completing this project, you will gain hands-on experience with Android Studio, Compose libraries, and database connectivity, enabling you to design functional and user-friendly Android applications.




# Code:

```
import android.annotation.SuppressLint
import android.content.Context
import android.os.Bundle
import android.widget.Toast
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.annotation.DrawableRes
import androidx.annotation.StringRes
import androidx.compose.foundation.Image
import androidx.compose.foundation.background
import androidx.compose.foundation.layout.
import androidx.compose.foundation.shape.CircleShape
import androidx.compose.foundation.shape.RoundedCornerShape
import androidx.compose.material.
import androidx.compose.material.icons.Icons
import androidx.compose.material.icons.filled.
import androidx.compose.runtime.Composable
```



```
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.draw.clip
import androidx.compose.ui.graphics.Color
import androidx.compose.foundation.lazy.LazyColumn
import androidx.compose.foundation.lazy.items
import androidx.compose.material.Text
import androidx.compose.ui.unit.dp
import androidx.compose.ui.graphics.RectangleShape
import androidx.compose.ui.layout.ContentScale
import androidx.compose.ui.platform.LocalContext
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.res.stringResource
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.unit.sp
import androidx.core.content.ContextCompat.startActivity
import com.example.snackordering.ui.theme.SnackOrderingTheme
import android.content.Intent as Intent1
```




```

class MainPage : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            SnackOrderingTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {
                    FinalView(this)
                    val context = LocalContext.current
                    //PopularFoodColumn(context)
                }
            }
        }
    }
}

@Composable
fun TopPart() {
    Row(
        modifier = Modifier

```



```
.fillMaxWidth()

.background(Color(0xffeceef0)), Arrangement.SpaceBetween
){
    Icon(
        imageVector = Icons.Default.Add, contentDescription = "Menu Icon",
        Modifier
            .clip(CircleShape)
            .size(40.dp),
        tint = Color.Black,
    )
    Column(horizontalAlignment = Alignment.CenterHorizontally) {
        Text(text = "Location", style = MaterialTheme.typography.subtitle1, color = Color.Black)
```



```
Row {
```

```
    Icon(
```

```
        imageVector = Icons.Default.LocationOn,
```

```
        contentDescription = "LOCATION",
```

```
        tint = Color.Red,
```

```
    )
```

```
    Text(text = "Coimbatore" , color = Color.Black)
```

```
}
```

```
}
```

```
Icon(
```

```
    imageVector = Icons.Default.Notifications, contentDescription = "Notification Icon",
```

```
    Modifier
```



```
        .size(40.dp),  
        tint = Color.Black,  
    )  
}  
  
@Composable  
fun CardPart() {  
    Card(modifier = Modifier.size(width = 310.dp, height = 150.dp), RoundedCornerShape(20.dp)) {  
        Row(modifier = Modifier.padding(10.dp), Arrangement.SpaceBetween) {  
            Column(verticalArrangement = Arrangement.spacedBy(12.dp)) {  
                Text(text = "Special Discounts")  
                Text(text = "up to 50%", style = MaterialTheme.typography.h5)  
                Button(onClick = {}, colors = ButtonDefaults.buttonColors(Color.White)) {
```






```
        Text(text = "Claim voucher", color = MaterialTheme.colors.surface)
    }
}


Image(
    painter = painterResource(id = R.drawable.burger),
    contentDescription = "Food Image", Modifier.size(width = 100.dp, height = 200.dp)
)
}

}

@Composable
fun PopularFood(
    @DrawableRes drawable: Int,
```




```
@StringRes text1: Int,  
context: Context  
) {  
    Card(  
        modifier = Modifier  
            .padding(top=20.dp, bottom = 20.dp, start = 65.dp)  
            .width(250.dp)  
    ) {  
        Column(  
            verticalArrangement = Arrangement.Top,  
            horizontalAlignment = Alignment.CenterHorizontally  
        ) {
```



```
Spacer(modifier = Modifier.padding(vertical = 5.dp))  
Row(  
    modifier = Modifier  
        .fillMaxWidth(0.7f), Arrangement.End  
    ) {  
    Icon(  
        imageVector = Icons.Default.Star,  
        contentDescription = "Star Icon",  
        tint = Color.Red  
    )  
    Text(text = "4.6", fontWeight = FontWeight.Black)  
}
```

  
**Image(****painter = painterResource(id = drawable),****contentDescription = "Food Image",****contentScale = ContentScale.Crop,****modifier = Modifier****.size(100.dp)****.clip(CircleShape)****)****Text(text = stringResource(id = text1), fontWeight = FontWeight.Bold)****Row(modifier = Modifier.fillMaxWidth(0.7f), Arrangement.SpaceBetween) {****/\*TODO Implement Prices for each card\*/****Text(**



```
        text = "₹100",  
        style = MaterialTheme.typography.h6,  
        fontWeight = FontWeight.Bold,  
        fontSize = 18.sp  
    )  
    IconButton(onClick = {  
        //var no=FoodList.lastIndex;  
        //Toast.  
        val intent = Intent1(context, TargetActivity::class.java)  
        context.startActivity(intent)  
    }) {  
        Icon(  
            imageVector = Icons.Default.ShoppingCart,
```




```
        contentDescription = "shopping cart",  
    )  
}  
}  
}  
}
```

```
private val FoodList = listOf(  
    R.drawable.sandwich to R.string.sandwich,  
    //R.drawable.sandwich to R.string.burgers,  
    R.drawable.pack to R.string.pack,  
    R.drawable.pasta to R.string.pasta,  
).map { DrawableStringPair(it.first, it.second) }
```



```
        R.drawable.tequila to R.string.tequila,  
        R.drawable.wine to R.string.wine,  
        R.drawable.salad to R.string.salad,  
        R.drawable.pop to R.string.popcor  
    ).map { DrawableStringPair(it.first, it.second) }  
  
private data class DrawableStringPair(  
    @DrawableRes val drawable: Int,  
    @StringRes val text1: Int  
)  
  
@Composable  
fun App(context: Context) {
```



```
Column(  
    modifier = Modifier  
        .fillMaxSize()  
        .background(Color(0xffeceef0))  
        .padding(10.dp),  
    verticalArrangement = Arrangement.Top,  
    horizontalAlignment = Alignment.CenterHorizontally  
) {  
    Surface(modifier = Modifier, elevation = 5.dp) {  
        TopPart()  
    }  
    Spacer(modifier = Modifier.padding(10.dp))  
    CardPart()  
}
```





```
        Spacer(modifier = Modifier.padding(10.dp))

        Row(modifier = Modifier.fillMaxWidth(), Arrangement.SpaceBetween) {

            Text(text = "Recommended snacks", style = MaterialTheme.typography.h5, color = Color.Black)

            Text(text = "View all", style = MaterialTheme.typography.subtitle1, color = Color.Black)

        }

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

        PopularFoodColumn(context) // <- call the function with parentheses

    }

}

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



```
LazyColumn(  
    modifier = Modifier.fillMaxSize(),  
    content = {  
        items(FoodList) { item ->  
            PopularFood(context = context,drawable = item.drawable, text1 = item.text1)  
        }  
    },  
    verticalArrangement = Arrangement.spacedBy(16.dp))  
}
```



```
@SuppressLint("UnusedMaterialScaffoldPaddingParameter")
```

```
@Composable
```

```
fun FinalView(mainPage: MainPage) {
```

```
    SnackOrderingTheme {
```

```
        Scaffold() {
```

```
            val context = LocalContext.current
```

```
            App(context)
```

```
        }
```

```
    }
```

```
}
```



```
package com.example.snackordering
```

```
import android.icu.text.SimpleDateFormat
```

```
import android.os.Bundle
```

```
import android.util.Log
```

```
import androidx.activity.ComponentActivity
```

```
import androidx.activity.compose.setContent
```

```
import androidx.compose.foundation.Image
```


```
import androidx.compose.foundation.layout.*
```

```
import androidx.compose.foundation.lazy.LazyColumn
```


```
import androidx.compose.foundation.lazy.LazyRow
```

```
import androidx.compose.foundation.lazy.items
```

```
import androidx.compose.material.MaterialTheme
```



```
import androidx.compose.material.Surface
import androidx.compose.material.Text
import androidx.compose.runtime.Composable
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.layout.ContentScale
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp
import com.example.snackordering.ui.theme.SnackOrderingTheme
import java.util.*
```



```
class AdminActivity : ComponentActivity() {  
    private lateinit var orderDatabaseHelper: OrderDatabaseHelper  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        orderDatabaseHelper = OrderDatabaseHelper(this)  
        setContent {  
            SnackOrderingTheme {  
                // A surface container using the 'background' color from the theme  
                Surface(  
                    modifier = Modifier.fillMaxSize(),  
                    color = MaterialTheme.colors.background  
                ) {  
                    val data=orderDatabaseHelper.getAllOrders();
```



```
        Log.d("swathi" ,data.toString())

        val order = orderDatabaseHelper.getAllOrders()

        ListListScopeSample(order)
    }
}

}

}

}

@Composable
fun ListListScopeSample(order: List<Order>) {

    Image(

        painterResource(id = R.drawable.pink), contentDescription = "",

        alpha = 0.5F,

        contentScale = ContentScale.FillHeight)
```



```
Text(text = "Order Tracking", modifier = Modifier.padding(top = 24.dp, start = 106.dp, bottom = 24.dp ), color = Color.White, fontSize = 30.sp)
```

```
Spacer(modifier = Modifier.height(30.dp))
```

```
LazyRow(
```

```
    modifier = Modifier
```

```
        .fillMaxSize()
```

```
        .padding(top = 80.dp),
```

```
        horizontalArrangement = Arrangement.SpaceBetween
```

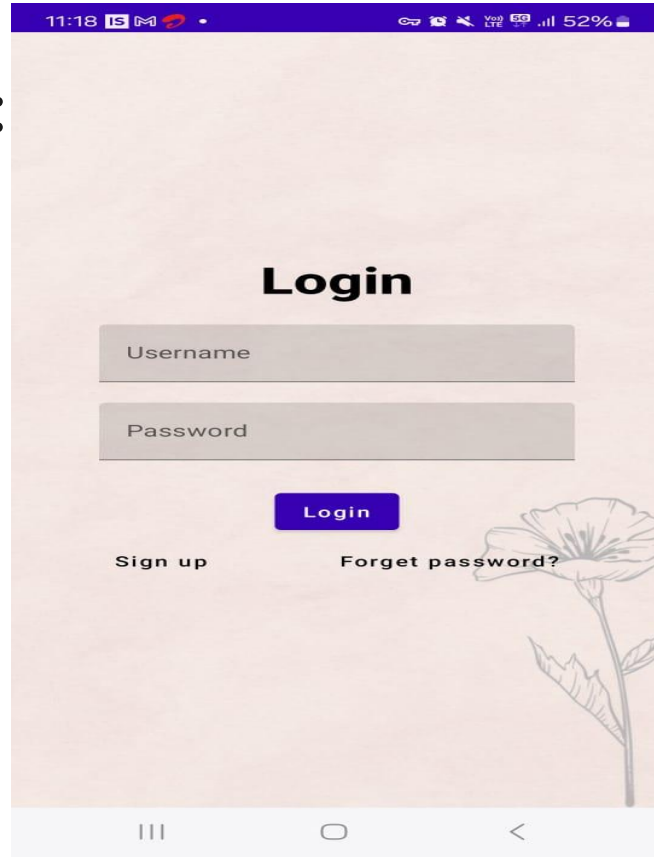
```
) {
```





```
item {  
    LazyColumn {  
        items(order) { order ->  
            Column(modifier = Modifier.padding(top = 16.dp, start = 48.dp, bottom = 20.dp)) {  
                Text("Quantity: ${order.quantity}")  
                Text("Address: ${order.address}")  
            }  
        }  
    }  
}
```

Output:





11:18 5G 52%

## Register


Username

Email

Password

Register

Have an account? [Log in](#)



III O <

