Snack Squad: A Snack Ordering and Delivery App

Final Project Report

Prepared For: Smart-Internz Andriod Guide Project

By: Aditya Abhinav Pol

Affiliation: D Y Patil Agriculture and Technical University, Talsande

Date: 20 August 2025

1. Abstract:

Snack Squad is an innovative and user-friendly mobile application designed to revolutionize the way people order and enjoy snacks. This project aims to provide users with a seamless and customizable snack ordering experience, ensuring convenience, variety, and satisfaction.

The application allows users to browse through an extensive menu of snacks, customize their orders according to personal preferences, and have them delivered right to their doorstep. With a focus on user experience, Snack Squad incorporates features such as real-time order tracking, personalized recommendations, and secure payment options.

Key functionalities include:

- ✓ **Customizable Orders:** Users can modify their snack orders to suit their tastes, including choosing ingredients, portion sizes, and dietary preferences.
- ✓ **Real-time Tracking:** The app provides real-time updates on order status, from preparation to delivery.
- ✓ **Personalized Recommendations:** Leveraging user data, Snack Squad offers tailored snack suggestions based on past orders and preferences.
- ✓ **Secure Payments:** Multiple secure payment gateways are integrated to ensure safe transactions.
- ✓ **User-friendly Interface:** The app's design prioritizes ease of use, ensuring a smooth and enjoyable ordering process.

Snack Squad aims to enhance the snacking experience by combining convenience with personalization, catering to the diverse needs and cravings of snack enthusiasts. Through continuous innovation and user feedback, the app strives to become the go-to platform for snack ordering and delivery.

By addressing common pain points in snack ordering and focusing on user satisfaction, Snack Squad sets a new standard in the industry. Our goal is to make snacking not just convenient, but an enjoyable and personalized experience. Join us as we redefine how snacks are ordered and enjoyed.

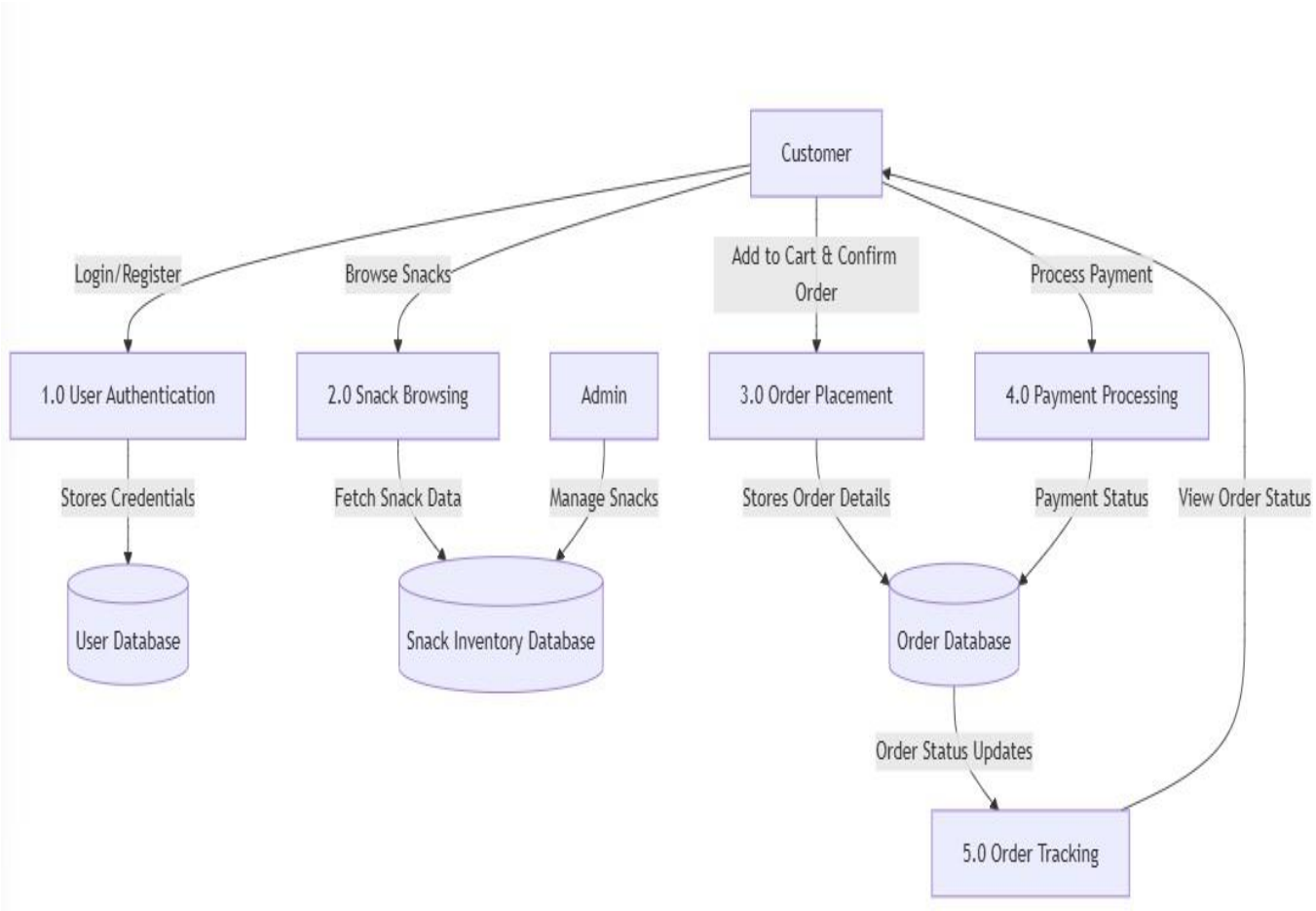
2. Introduction:

In today's fast-paced world, convenience and personalization have become key aspects of everyday life, especially in how we access our favorite treats. Snack Squad emerges as a pioneering solution to meet these demands, providing a customizable snack ordering and delivery platform that caters to individual preferences and dietary needs. Designed with user experience at its core, Snack Squad aims to transform the snacking landscape by offering a wide variety of snacks that can be tailored to suit each user's taste.

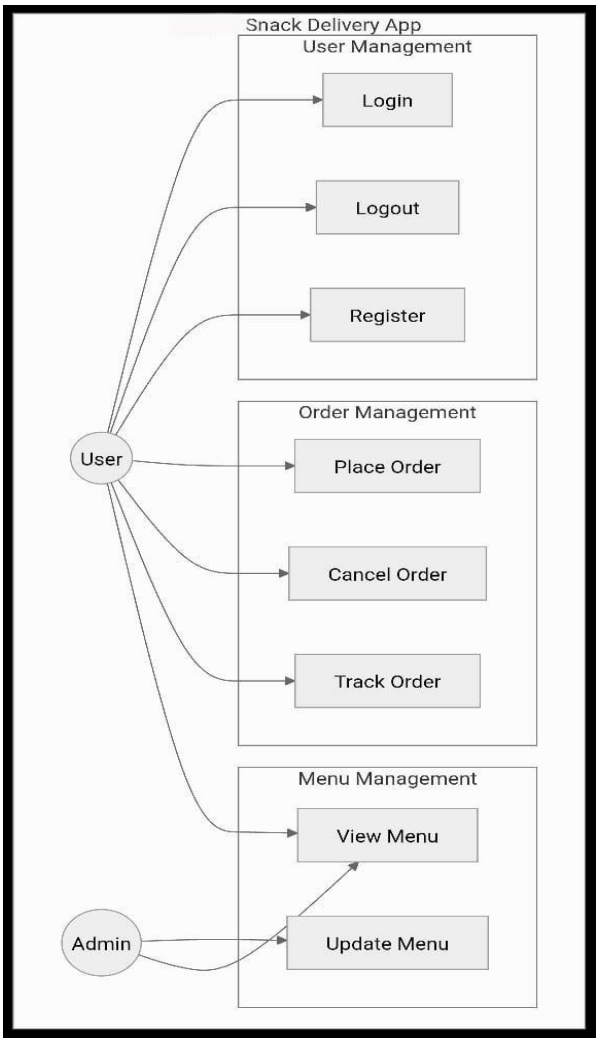
By leveraging cutting-edge technology, Snack Squad not only simplifies the process of ordering snacks but also enhances it with features such as real-time order tracking, personalized recommendations, and secure payment options. Whether you're craving a mid-day snack or planning for a movie night, Snack Squad ensures that your favorite snacks are just a few taps away, delivered straight to your doorstep.

As the snacking industry continues to evolve, Snack Squad stands out by addressing common pain points and focusing on user satisfaction. Join us in redefining the snacking experience with a blend of convenience, variety, and customization. Snack Squad is more than just an app; it's your personal snack concierge, ready to satisfy your cravings at any time.

3. Data flow diagram:



4. Use case diagram:



5. Software Requirement:

a) Operating System:

- Compatible with Android platforms.

b) Hardware Requirements:

- Android: Devices with at least 2GB of RAM and a quad-core processor.

c) Software Requirements:

- Programming Languages: Java/Kotlin for Android development
- Version Control: Git for managing source code and collaboration among developers.

6. Sample Program Code :

```
package  com.example.snackordering

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
                ) {
```

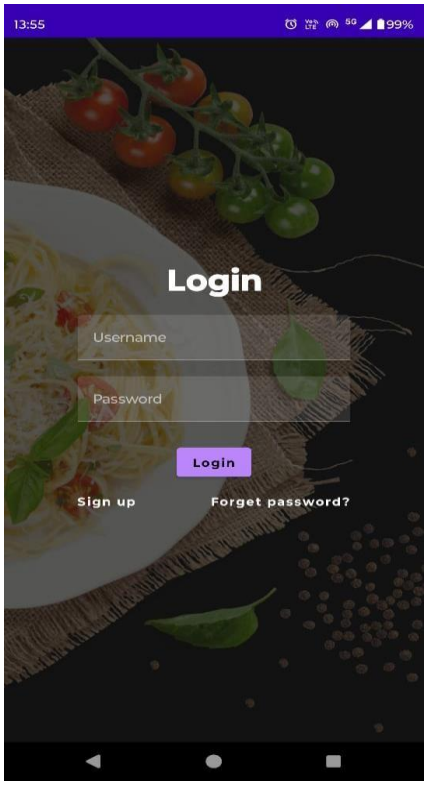
```
                FinalView(this)
                val context = LocalContext.current
                //PopularFoodColumn(context)
            }
        }
    }
}

@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 = "Accra" , color = Color.Black)
            }

        }
        Icon(
            imageVector = Icons.Default.Notifications, contentDescription = "Notification Icon",

            Modifier
                .size(45.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)) {
```

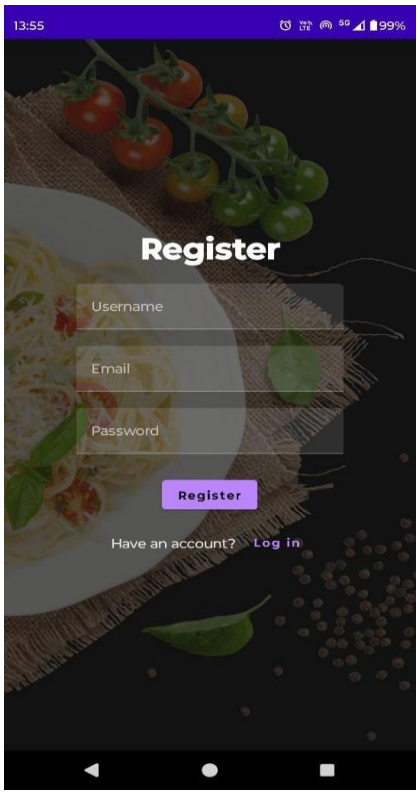
```
            Text(text = "Get Special Discounts")
            Text(text = "up to 85%", style = MaterialTheme.typography.h5)
            Button(onClick = {}, colors = ButtonDefaults.buttonColors(Color.White)) {
                Text(text = "Claim voucher", color = MaterialTheme.colors.surface)
            }
        }
        Image(
            painter = painterResource(id = R.drawable.food_tip_im),
            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.Yellow
                )
                Text(text = "4.3", 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) {
```

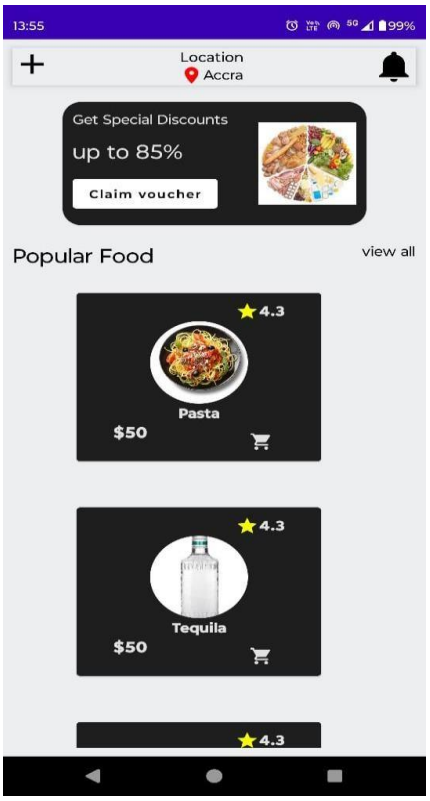
```
                /*TODO Implement Prices for each card*/
                Text(
                    text = "$50",
                    style = MaterialTheme.typography.h6,
                    fontWeight = FontWeight.Bold,
                    fontSize = 18.sp
                )

                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.sandwish to R.string.sandwich,
    R.drawable.sandwish to R.string.burgers,
    R.drawable.pack to R.string.pack,
    R.drawable.pasta to R.string.pasta,
    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.popcorn
).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
```

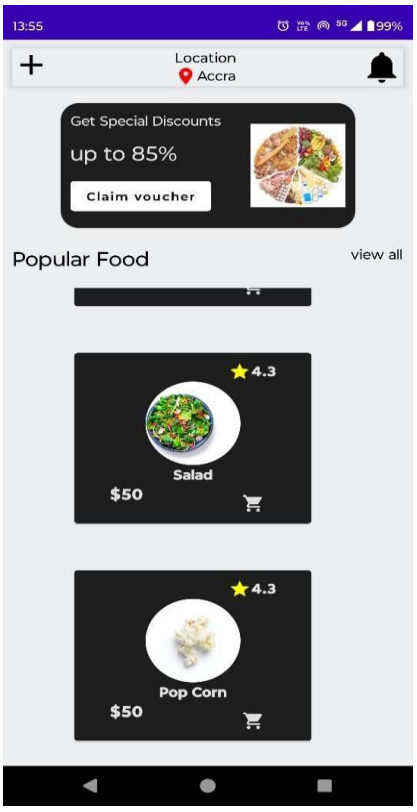
```
    ) {
        Surface(modifier = Modifier, elevation = 5.dp) {
            TopPart()
        }
        Spacer(modifier = Modifier.padding(10.dp))
        CardPart()

        Spacer(modifier = Modifier.padding(10.dp))
        Row(modifier = Modifier.fillMaxWidth(), Arrangement.SpaceBetween) {
            Text(text = "Popular Food", 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)
                abstract class Context
            }
        },
        verticalArrangement = Arrangement.spacedBy(16.dp))
}

@SuppressLint("UnusedMaterialScaffoldPaddingParameter")
@Composable
fun FinalView(mainPage: MainPage) {
    SnackOrderingTheme {
        Scaffold()  {
            val context = LocalContext.current
            App(context)
        }
    }
}
```

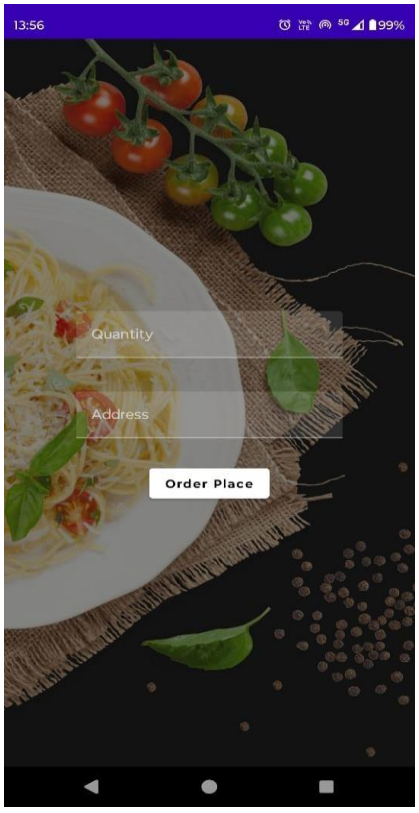
7. Output

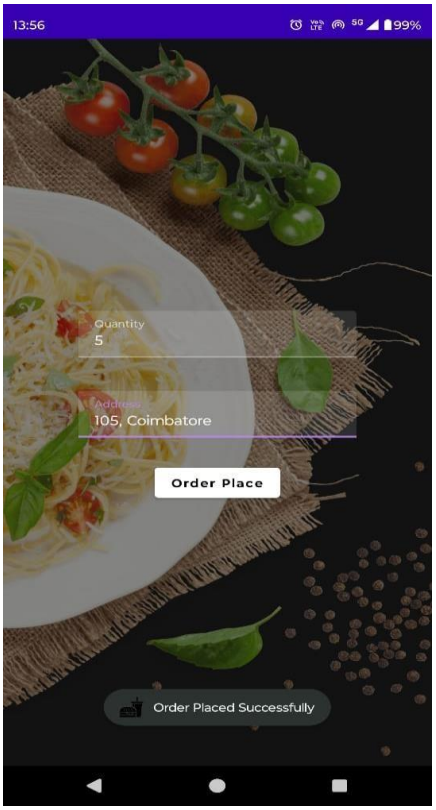

13:55
99%
Login
Username
Password
Login
Sign up
Forget password?


13:55
99%
Register
Username
Email
Password
Register
Have an account? Log in


13:55
99%
Location
Accra
Get Special Discounts
up to 85%
Claim voucher
Popular Food
view all
4.3
Pasta
$50
4.3
Tequila
$50
4.3


13:55
99%
Location
Accra
Get Special Discounts
up to 85%
Claim voucher
Popular Food
view all
4.3
Salad
$50
4.3
Pop Corn
$50


13:56
99%
Quantity
Address
Order Place


13:56
99%
Quantity
5
Address
105, Coimbatore
Order Place
Order Placed Successfully

8. Conclusion:

The Snack Squad application represents a significant advancement in the convenience and personalization of snack ordering. Through its innovative features, such as customizable orders, real-time tracking, and secure payment processing, Snack Squad not only meets the current demands of snack enthusiasts but also sets a new benchmark for user satisfaction in the food delivery industry. By leveraging cutting-edge technology and focusing on a seamless user experience, Snack Squad ensures that users can enjoy their favorite snacks with minimal hassle and maximum satisfaction. Our commitment to quality, security, and continuous improvement underpins the development and deployment of this application, making Snack Squad a pioneering solution in the market.

9. Future Enhancement:

To continually improve and adapt to user needs, several enhancements are planned for the future development of Snack Squad:

- Enhanced Personalization: Incorporating machine learning algorithms to better understand user preferences and provide more accurate recommendations.
- Expanded Snack Variety: Partnering with more local and international snack vendors to offer a wider range of options.
- Subscription Services: Introducing subscription-based snack delivery plans for regular users, providing convenience and cost savings.
- Advanced Order Customization: Adding more options for order customization, such as allergen filters and detailed nutritional information.
- Loyalty Programs: Implementing a rewards system to incentivize repeat orders and enhance user engagement.
- Voice Ordering: Integrating with voice assistants to allow users to place orders via voice commands for added convenience.
- Sustainability Initiatives: Partnering with eco-friendly delivery services and offering sustainable packaging options to minimize environmental impact.
- Improved Security Measures: Continuously updating security protocols to protect user data and ensure safe transactions.
- Global Expansion: Scaling the app to support multiple languages and currencies, enabling Snack Squad to serve users worldwide.