THE Easy Food

INTRODUCTION

Purpose

With the improvement of technology, online food ordering systems are becoming a popular topic. That's because they are serving the increasing demand for convince. The main purpose of an online ordering system is to provide customers for a way to place an order at a restaurant over the internet.

The main reason is that it benefits both the customer and the business . With a website or mobile app, customers can easily browse all the dishes the restaurant has available, customize dishes to their requirements and place an order. It can also save their favourite orders allowing them to easily re-order that in the future. The target market also used to be busy people who needed a restaurant delivery service because they didn’t have time. Now, the target market is everyone. Most people now require home deliveries.

With a number of restaurants opening in every nook and corner of cities which have brought in a number of international and national cuisines, people have been looking forward to trying their food. Not everyone can go and dine at a restaurant, hence, the[**on-demand food delivery app**](https://www.goodfirms.co/company/octal-it-solution)development came into being, which not only made it easy for the customers to enjoy scrumptious food at the leisure of their home but also offered a wide variety of cuisines from across the globe.

Project Scope

The Easy Food app can sale Food product, preferred brands, kitchen needs, essential restaurant supplies and more, through this online, one stop Food store. It provides you with a convenient way to sale from your Food shopping app. You can use this app as one big super market app to sale product of your store. This app make easy for user to buy product from store with easy steps and store can get easy order.

Features

The following are the features of the easy food are :

➢ Native Android Application

➢ Classified Products

➢ Easy add to cart with one click item

➢ Manage Delivery Address Street wise

➢ Choose Delivery time slot

➢ User Menu

• Manage Orders

• Manage Profile

• Manage Address

**SYSTEM ANALYSIS**

Hardware Used:

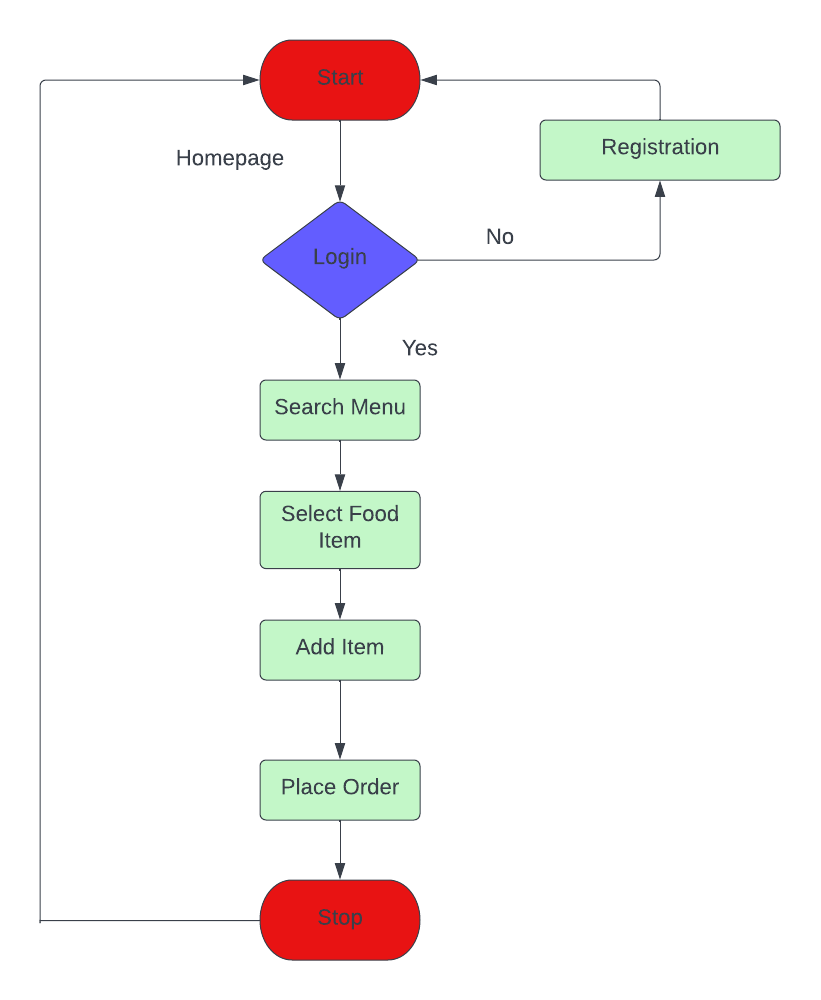
* Processor Intel i5 8th Gen
* Ram 8GB
* Monitor
* Keyboard and Mouse

Software Used:

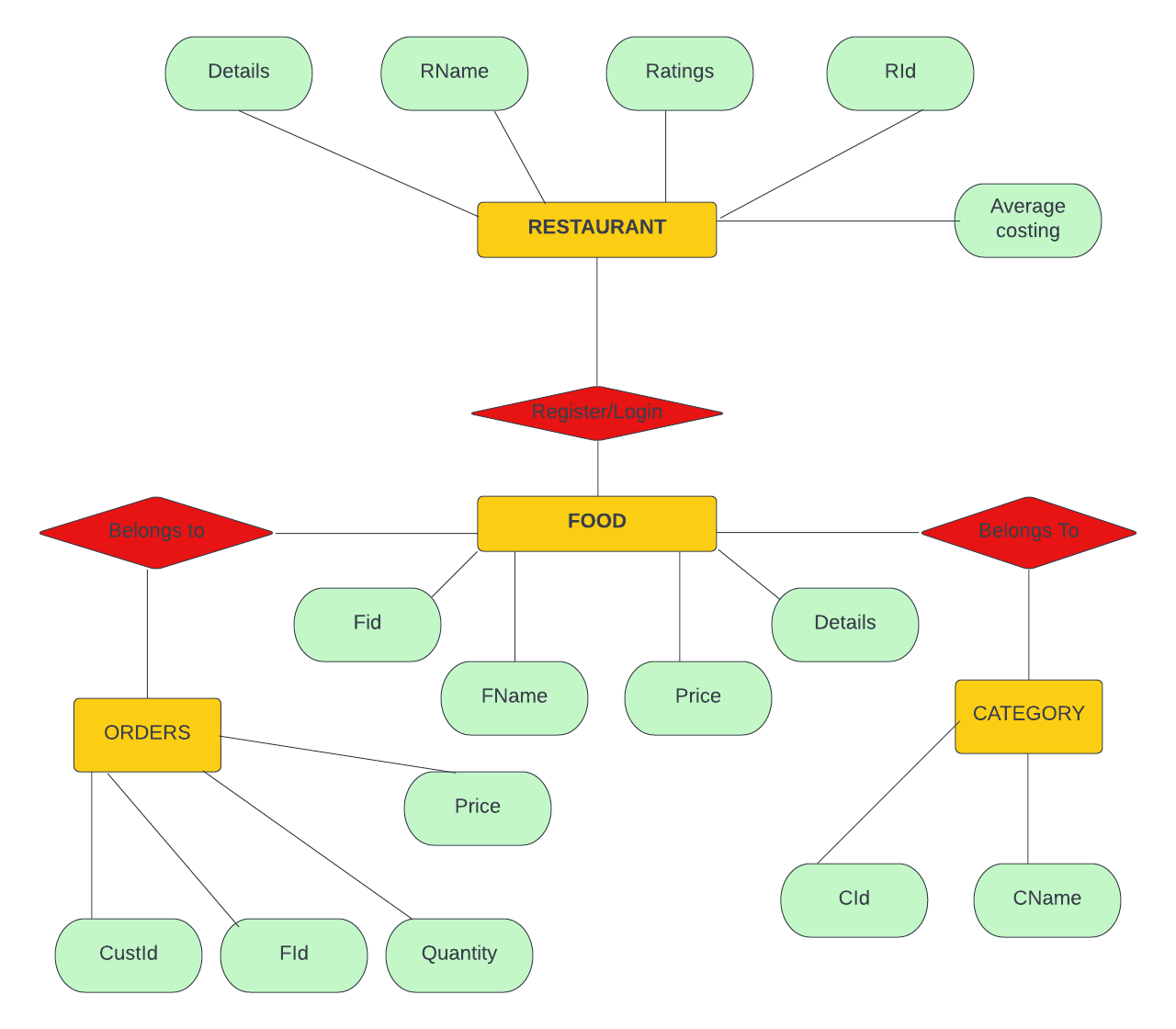
* Operating System - Windows 10
* Developing Tool - Netbeans IDE
* Language - Java
* Database - SQLite

**SYSTEM DESIGN AND SPECIFICATIONS**

**Flow Chart**



ER DIAGRAM

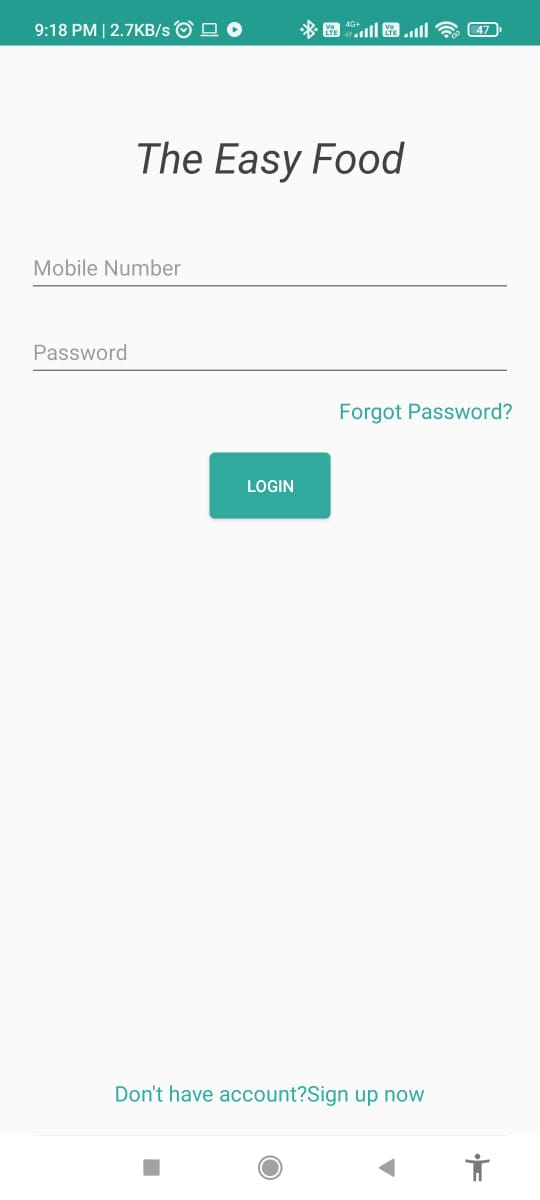
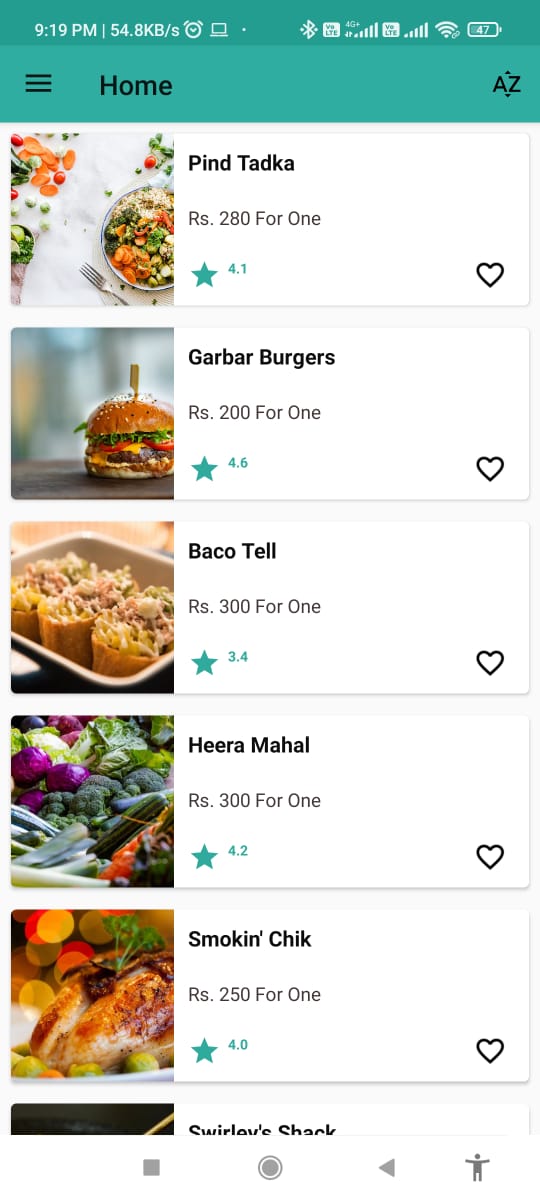
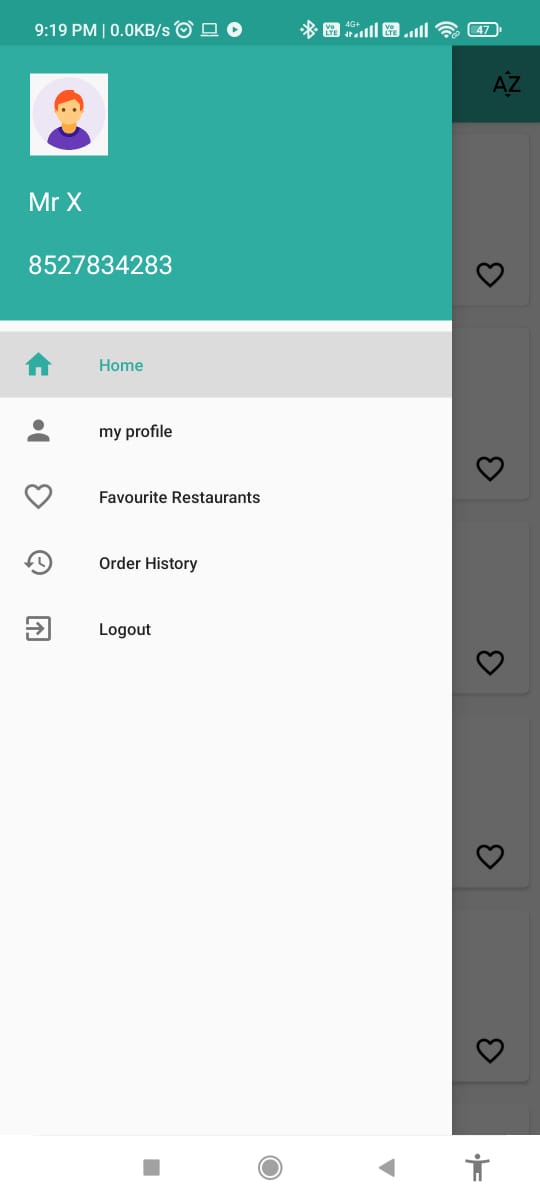


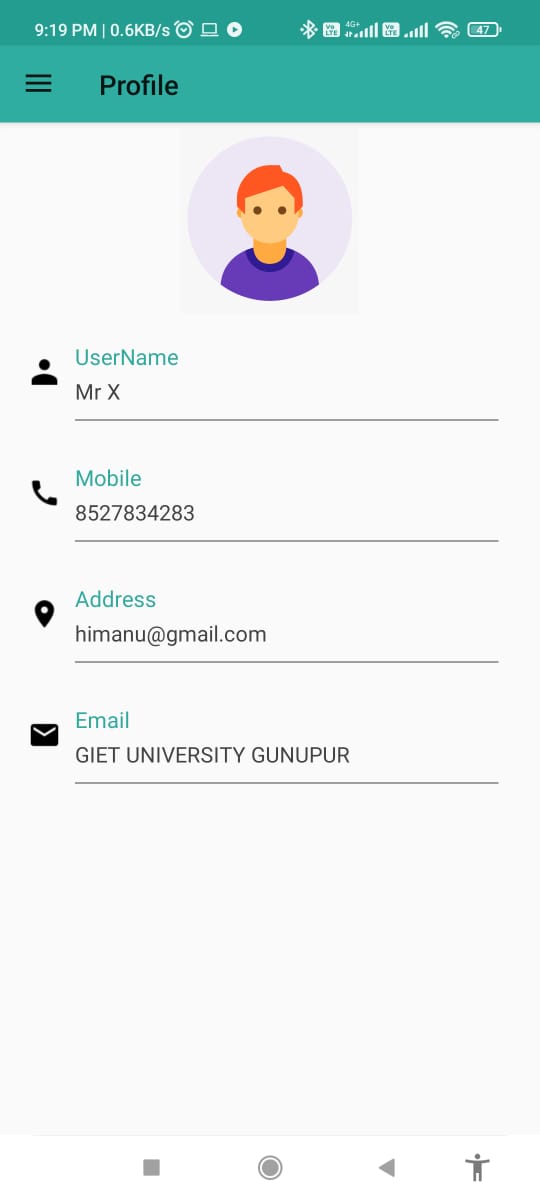
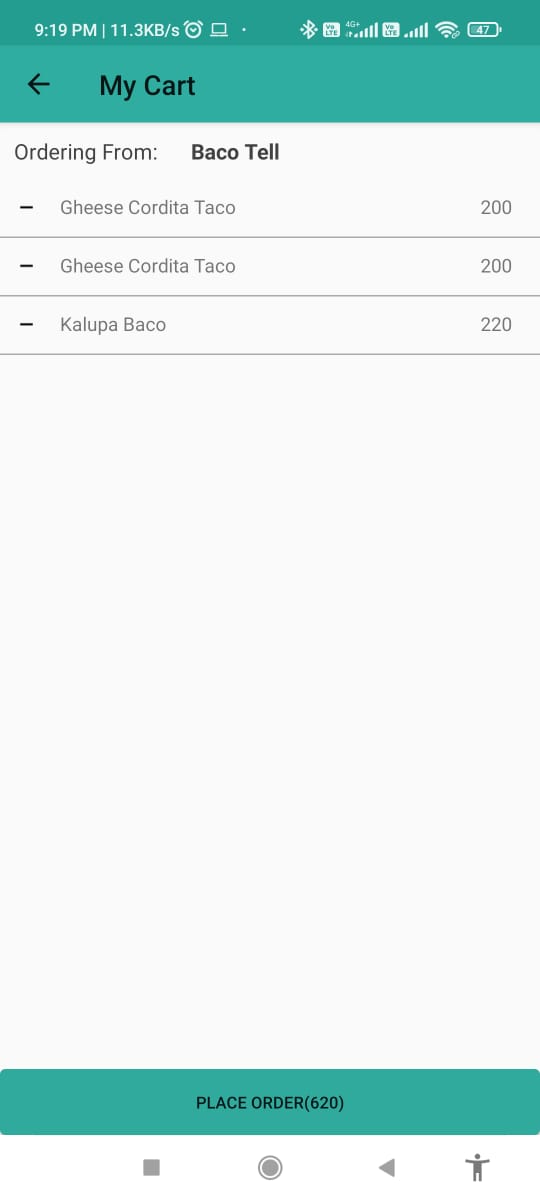
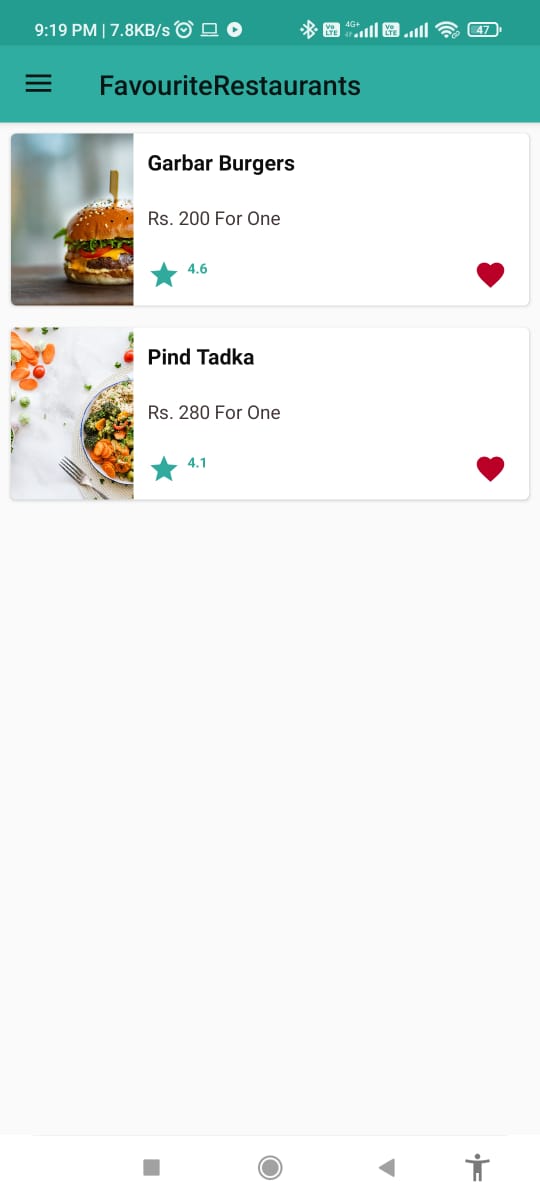
Low Level Design

* **Algorithm:**

1. Start
2. Sign Up
3. Login
4. Home
   1. Main Menu
   2. Menu List
   3. Manage Orders
   4. Favourites
   5. Sort By
   6. Orders History
   7. My Profile
5. Logout
6. Stop

Screen Shot Diagram

CODING

Main Activity

package com.basics.fooddeliveryapp.Activity  
  
import android.content.Context  
import android.content.Intent  
import android.content.SharedPreferences  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.MenuItem  
import android.widget.FrameLayout  
import android.widget.TextView  
import androidx.appcompat.app.ActionBarDrawerToggle  
import androidx.appcompat.app.AlertDialog  
import androidx.appcompat.widget.Toolbar  
import androidx.coordinatorlayout.widget.CoordinatorLayout  
import androidx.core.view.GravityCompat  
import androidx.drawerlayout.widget.DrawerLayout  
import com.basics.fooddeliveryapp.ProfileActivity.LoginActivity  
import com.basics.fooddeliveryapp.Fragment.\*  
import com.basics.fooddeliveryapp.R  
import com.google.android.material.navigation.NavigationView  
  
class MainActivity : AppCompatActivity() {  
 lateinit var drawerLayout: DrawerLayout  
 lateinit var frameLayout: FrameLayout  
 lateinit var toolbar: Toolbar  
 lateinit var navigationView: NavigationView  
 lateinit var coordinatorLayout: CoordinatorLayout  
  
 lateinit var sharedPrefrence: SharedPreferences  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
  
 drawerLayout = findViewById(R.id.*drawerLayout*)  
 frameLayout = findViewById(R.id.*frame*)  
 toolbar = findViewById(R.id.*toolbar*)  
 navigationView = findViewById(R.id.*navigationLayout*)  
 coordinatorLayout = findViewById(R.id.*coordinatorLayout*)  
 sharedPrefrence = getSharedPreferences("Login Preference", Context.*MODE\_PRIVATE*)  
 setUpToolbar()  
 openHome()  
  
  
 val actionBarDrawerToggle = ActionBarDrawerToggle(  
 this@MainActivity,  
 drawerLayout,  
 R.string.*open\_drawer*,  
 R.string.*closed\_drawer* )  
  
 drawerLayout.addDrawerListener(actionBarDrawerToggle)  
 actionBarDrawerToggle.syncState()  
  
 navigationView.setNavigationItemSelectedListener **{** when (**it**.*itemId*) {  
  
 R.id.*home* -> {  
 openHome()  
 drawerLayout.closeDrawers()  
  
  
 }  
  
 R.id.*myProfile* -> {  
 *supportFragmentManager*.beginTransaction()  
 .replace(  
 R.id.*frame*,  
 ProfileFragment()  
 )  
 .commit()  
  
 *supportActionBar*?.*title* = "Profile"  
 drawerLayout.closeDrawers()  
  
  
 }  
  
 R.id.*favouriteRestaurants* -> {  
 *supportFragmentManager*.beginTransaction()  
 .replace(  
 R.id.*frame*,  
 FavouriteRestaurantsFragment()  
 )  
 .commit()  
 *supportActionBar*?.*title* = "FavouriteRestaurants"  
 drawerLayout.closeDrawers()  
  
  
 }  
  
 R.id.*orderHistory* -> {  
 *supportFragmentManager*.beginTransaction()  
 .replace(  
 R.id.*frame*,  
 OrderHistoryFragment()  
 )  
 .commit()  
 *supportActionBar*?.*title* = "Order History"  
 drawerLayout.closeDrawers()  
  
 }  
  
 */\* R.id.faqs -> {  
 supportFragmentManager.beginTransaction()  
 .replace(  
 R.id.frame,  
 FaqsFragment()  
 )  
 .commit()  
 supportActionBar?.title = "Faqs"  
 drawerLayout.closeDrawers()  
 }\*/* R.id.*logout* -> {  
 drawerLayout.closeDrawers()  
 var alertDialog = AlertDialog.Builder(this)  
 alertDialog.setTitle("Confirmation")  
 alertDialog.setMessage("Are your sure you want to log out?")  
  
 alertDialog.setPositiveButton("Yes") **{** text, listener **->** val intent = Intent(  
 this,  
 LoginActivity::class.*java* )  
 val sharedPreferences =  
 getSharedPreferences("Login Preference", Context.*MODE\_PRIVATE*)  
  
 val editor = sharedPreferences.edit()  
  
 editor.clear()  
 editor.commit()  
  
 startActivity(intent)  
 finishAffinity()  
  
 **}** alertDialog.setNegativeButton("NO") **{** text, listener **->** openHome()  
 navigationView.setCheckedItem(R.id.*home*)  
 **}** alertDialog.create()  
 alertDialog.show()  
  
 }  
  
  
 }  
 return@setNavigationItemSelectedListener true  
 **}** val headerView = navigationView.getHeaderView(0)  
  
 val txtUsername: TextView = headerView.findViewById(R.id.*txtUserName*)  
 val txtMobileNo: TextView = headerView.findViewById(R.id.*txtMobileNo*)  
  
 txtUsername.*text* = sharedPrefrence.getString("name", "username")  
 txtMobileNo.*text* = sharedPrefrence.getString("mobile", "+919191919191")  
  
  
 }  
  
 private fun setUpToolbar() {  
 setSupportActionBar(toolbar)  
 *supportActionBar*?.*title* = "Toolbar title"  
 *supportActionBar*?.setHomeButtonEnabled(true)  
 *supportActionBar*?.setDisplayHomeAsUpEnabled(true)  
 }  
  
 override fun onOptionsItemSelected(item: MenuItem): Boolean {  
  
  
 val id = item.*itemId* if (id == android.R.id.*home*) {  
 drawerLayout.openDrawer(GravityCompat.*START*)  
 }  
  
  
 return super.onOptionsItemSelected(item)  
 }  
  
 private fun openHome() {  
  
 val transaction = *supportFragmentManager*.beginTransaction()  
 val fragment = HomeFragment()  
 transaction  
 .replace(R.id.*frame*, fragment)  
 .commit()  
  
 *supportActionBar*?.*title* = "Home"  
  
 }  
  
 override fun onBackPressed() {  
 val frag = *supportFragmentManager*.findFragmentById(R.id.*frame*)  
  
 when (frag) {  
 !is HomeFragment -> {openHome()  
 navigationView.setCheckedItem(R.id.*home*)  
 }  
 else -> super.onBackPressed()  
  
 }  
 }  
  
  
}

My Cart Activity

package com.basics.fooddeliveryapp.Activity  
  
import android.content.Context  
import android.content.Intent  
import android.content.SharedPreferences  
import android.os.Bundle  
import android.view.MenuItem  
import android.widget.Button  
import android.widget.TextView  
import android.widget.Toast  
import androidx.appcompat.app.AppCompatActivity  
import androidx.appcompat.widget.Toolbar  
import androidx.recyclerview.widget.DividerItemDecoration  
import androidx.recyclerview.widget.LinearLayoutManager  
import androidx.recyclerview.widget.RecyclerView  
import com.android.volley.Response  
import com.android.volley.toolbox.JsonObjectRequest  
import com.android.volley.toolbox.Volley  
import com.basics.fooddeliveryapp.Adapter.CartAdapter  
import com.basics.fooddeliveryapp.R  
import org.json.JSONArray  
import org.json.JSONException  
import org.json.JSONObject  
  
  
class MyCartActivity : AppCompatActivity() {  
 lateinit var recyclerView: RecyclerView  
 lateinit var layoutManager: LinearLayoutManager  
 lateinit var cartAdapter: CartAdapter  
 lateinit var txtRestName: TextView  
 lateinit var sharedPreferences: SharedPreferences  
 lateinit var userSharedPreferences: SharedPreferences  
 lateinit var btnPlaceOrder: Button  
 lateinit var toolbar: Toolbar  
 private val foodId = *arrayListOf*<String>()  
  
 private val foodList = *arrayListOf*<HashMap<String, String>>()  
  
 var totalCost = 0  
 private var userId: String? = null  
 private lateinit var restaurantId: String  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_my\_cart*)  
  
 txtRestName = findViewById(R.id.*txtRestName*)  
 recyclerView = findViewById(R.id.*cartRecyclerView*)  
 layoutManager = LinearLayoutManager(this)  
 btnPlaceOrder = findViewById(R.id.*btnPlaceOrder*)  
 toolbar = findViewById(R.id.*toolbar*)  
 setUpToolbar()  
  
  
  
 sharedPreferences = getSharedPreferences("Cart Preference", Context.*MODE\_PRIVATE*)  
 userSharedPreferences = getSharedPreferences("Login Preference", Context.*MODE\_PRIVATE*)  
 userId = userSharedPreferences.getString("user\_id", null)  
  
  
 val allEntry = sharedPreferences.*all* if (*intent* != null) {  
 txtRestName.*text* = *intent*.getStringExtra("restaurant")  
 restaurantId = *intent*.getStringExtra("id")  
 }  
  
  
 allEntry.*forEach* **{** foodId.add(**it**.key)  
 **}** for (i in 0 *until* foodId.size) {  
 val values = sharedPreferences.getString(foodId[i], null)?.*split*("\n")  
  
 if (values != null) {  
 val name = values[0].*trim*()  
 val price = values[1].*trim*()  
  
 val map = HashMap<String, String>()  
 map["name"] = name  
 map["price"] = price  
 totalCost += try {  
 price.*toInt*()  
 } catch (e: ClassCastException) {  
 0  
 }  
  
  
 foodList.add(map)  
 cartAdapter = CartAdapter(this, foodList)  
  
 recyclerView.*adapter* = cartAdapter  
 recyclerView.*layoutManager* = layoutManager  
 recyclerView.addItemDecoration(  
 DividerItemDecoration(  
 recyclerView.*context*,  
 DividerItemDecoration.*VERTICAL* )  
 )  
  
  
 }  
 }  
  
 btnPlaceOrder.*text* = "Place Order($totalCost)"  
  
  
 btnPlaceOrder.setOnClickListener **{** val queue = Volley.newRequestQueue(this)  
  
 val url = "http://13.235.250.119/v2/place\_order/fetch\_result/"  
  
  
 val jsonObject = JSONObject()  
 jsonObject.put("user\_id", userId)  
 jsonObject.put("restaurant\_id", restaurantId)  
 jsonObject.put("total\_cost", totalCost.toString())  
  
 val jsonArray = JSONArray()  
  
  
 foodId.*forEach* **{** val foodJsonObject = JSONObject()  
  
 foodJsonObject.put("food\_item\_id", **it**)  
  
 jsonArray.put(foodJsonObject)  
 **}** jsonObject.put("food", jsonArray)  
  
  
 val jsonRequest =  
 object : JsonObjectRequest(Method.*POST*, url, jsonObject, Response.Listener **{** try {  
 val cartObject = **it**.getJSONObject("data")  
  
 val success = cartObject.getBoolean("success")  
 if (success) {  
 val intent = Intent(this, OrderPlacedActivity::class.*java*)  
 startActivity(intent)  
 finish()  
 } else {  
 Toast.makeText(this, "enter valid details", Toast.*LENGTH\_SHORT*).show()  
  
 }  
 } catch (e: JSONException) {  
 Toast.makeText(this, "json exception", Toast.*LENGTH\_SHORT*).show()  
  
 }  
 **}**, Response.ErrorListener **{** Toast.makeText(this, "some error occured", Toast.*LENGTH\_SHORT*).show()  
  
  
 **}**) {  
 override fun getHeaders(): MutableMap<String, String> {  
 val headers = HashMap<String, String>()  
 headers["Content-type"] = "application/json"  
 headers["token"] = "ec66f7766ff4a4"  
 return headers  
 }  
 }  
  
  
  
 queue.add(jsonRequest)  
 **}** }  
  
  
 private fun setUpToolbar() {  
 setSupportActionBar(toolbar)  
 *supportActionBar*?.*title* = "My Cart"  
 *supportActionBar*?.setHomeButtonEnabled(true)  
 *supportActionBar*?.setDisplayHomeAsUpEnabled(true)  
 }  
  
 override fun onOptionsItemSelected(item: MenuItem): Boolean {  
  
  
 val id = item.*itemId* if (id == android.R.id.*home*) {  
 finish()  
 }  
  
  
 return super.onOptionsItemSelected(item)

Conclusion

The project training undertaken by us at “Gandhi Institute of Engineering and Technology” helped me I not only gaining practical experience but also to adapt the organizational environments and procedures.

It is said that no system is perfect, Taking this into consideration, the project “ The Easy Food “ for “A6 Pvt. Ltd “ has been carried out successfully.

Thus we confidently conclude that this was not only beneficial in its technical and educational aspect but also wins equality in building up our personality.

BIBLIOGRAPHY

Android Appication with Kotlin (Hardik Trivedi)

Android Jam

Andriod Developer Documentation(developer.android.com/docs)

Udemy : Android Development : Kotlin-beginners