

Shehab Mohamed Ibrahim Khalil

18P7213

Group 1 Section 1

CESS Senior 2

Mobile Programming

Project Final Delivery

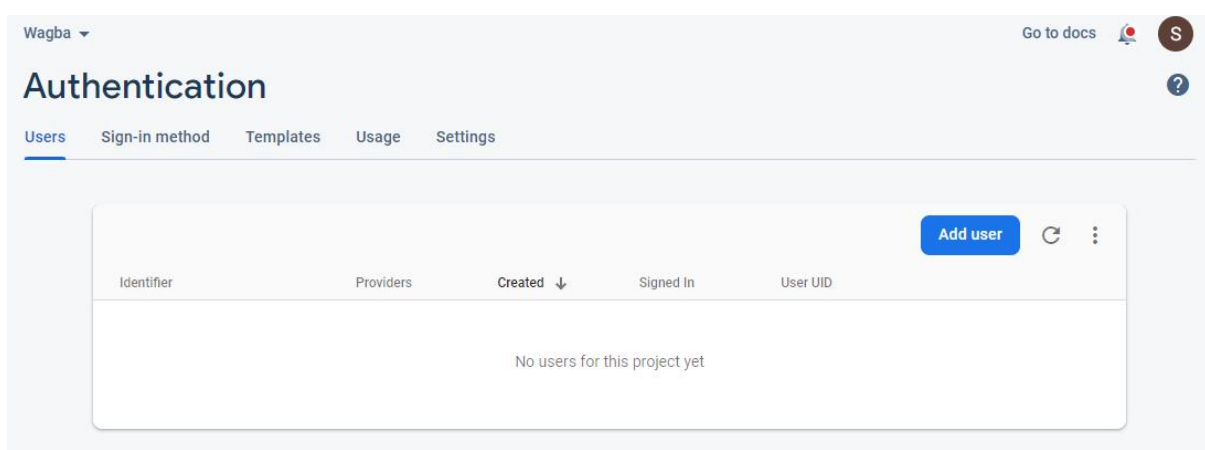
Test Account (Information provided should be an input in the Sign Up page):

- Email: mobile@project.com (or any other email)
 - Password: mobile (or any other password)
 - Full Name: Shehab Mohamed (or any other name)
 - Address: Ain Shams (or any other address)
-
- There are 2 applications for running the full functionality of the project, 1st app (named “Wagba”) is for the User to order food; 2nd app (named “WagbaHelper”) is to mimic the behavior of a restaurant (for tracking the state purpose).
 - Screenshots & the test case scenarios will of course be a mix of both apps during the Tracking Orders phase in the document.

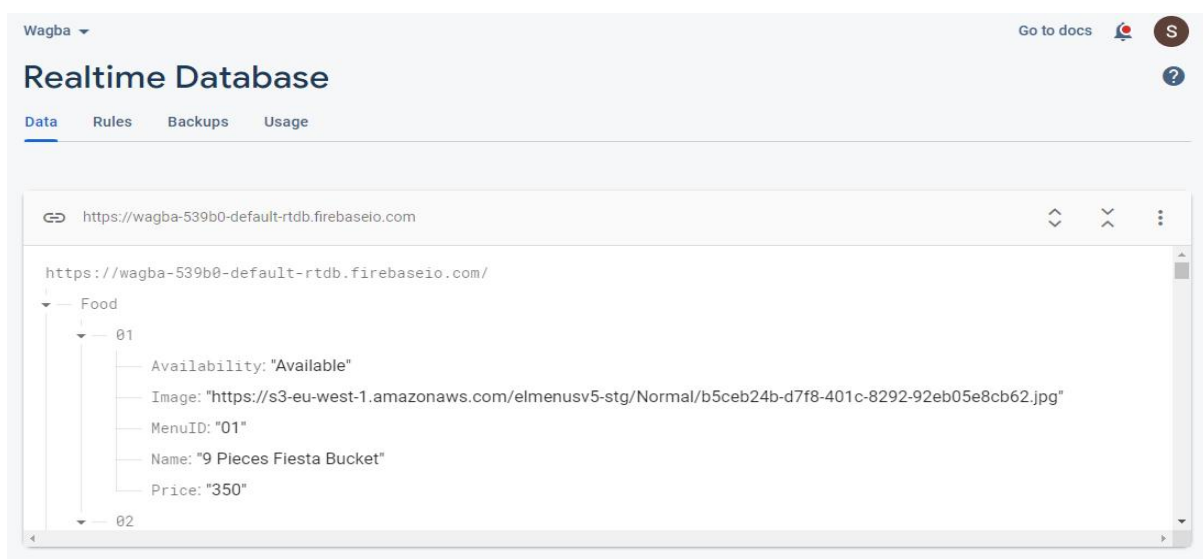
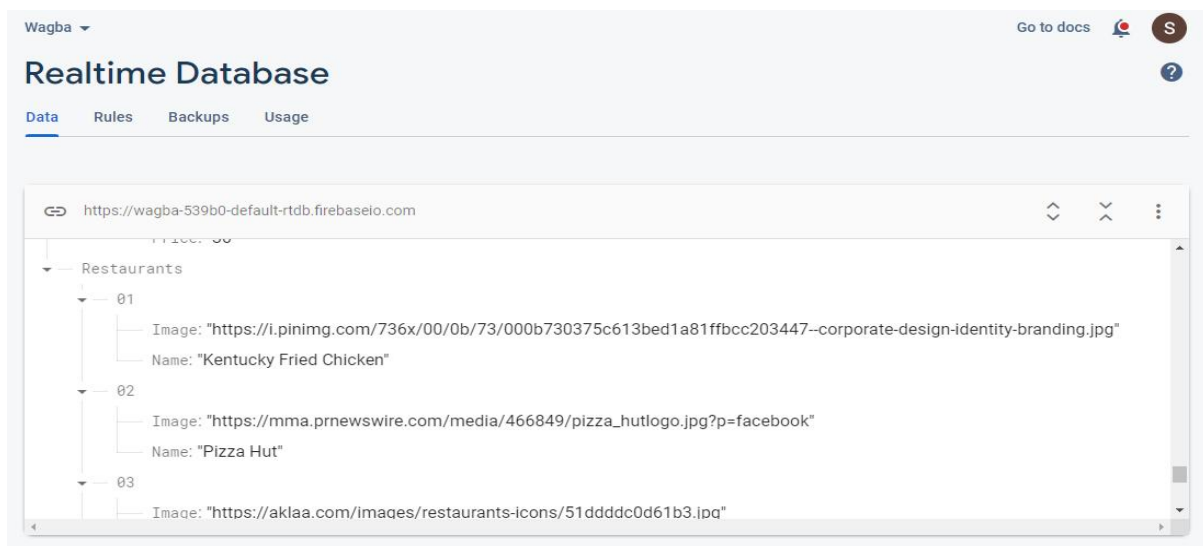
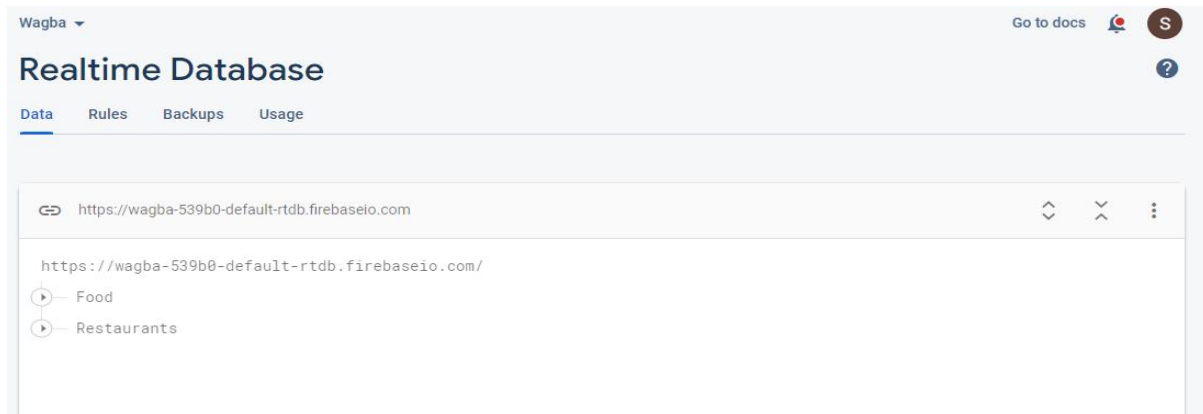
The following Screenshots accompanied with the full test case scenarios are in order of execution:

Firstly lets check what is in the Firebase before we run both apps:

No users signed up yet:



The Realtime Database does not contain any orders to be tracked yet, contents of the DB as of now is only the available restaurants and the available dishes:



Now let's see how the flow of Signing up and Signing in is carried on:
User's profile data like "Full Name" & "Address" gets stored in a local ROOM DB, so he has to provide these data while signing up, if he does not provide any one of them, the sign up operation will not be completed:

The image displays two sequential screenshots of a mobile application's 'SIGN UP' screen. The top status bar shows the time as 8:59 on the left and 9:00 on the right, with signal and battery icons on the right. The app interface features a white rounded rectangle with a blue border and the title 'SIGN UP' in blue. It contains four input fields: an email field with '18p7213@eng.asu.edu.eg', a password field with '*****', a 'Full Name' field, and an 'Address' field. A blue 'SIGN UP' button is at the bottom, with a link 'Already signed up? Log in!' below it. In the first screenshot, the 'Full Name' and 'Address' fields are empty, and a red error message 'Please enter your Name!' is shown over the 'Full Name' field. In the second screenshot, the 'Full Name' field is filled with 'Shehab Mohamed Ibrahim', but the 'Address' field is still empty, and a red error message 'Please enter your Address!' is shown over it. Both screenshots have a grey navigation bar at the bottom with three icons: a hamburger menu, a square, and a back arrow.

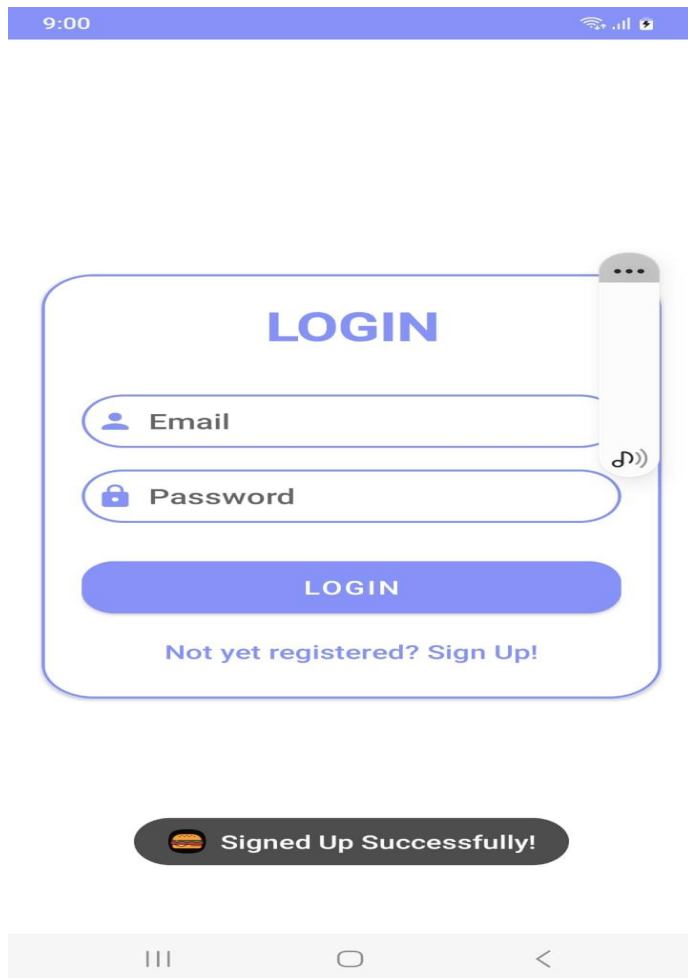
Email: 18p7213@eng.asu.edu.eg

Password: ainshams

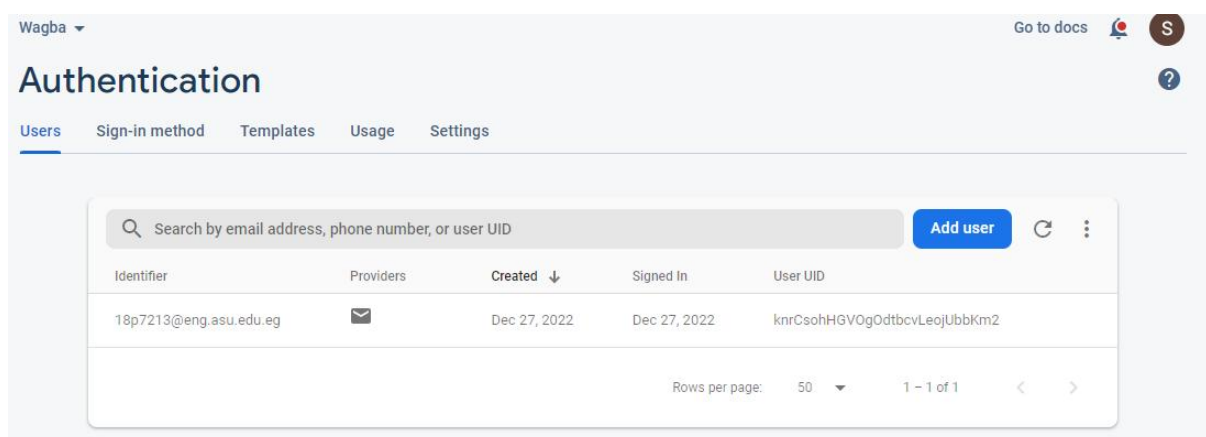
Full Name: Shehab Mohamed Ibrahim

Address: Ain Shams University

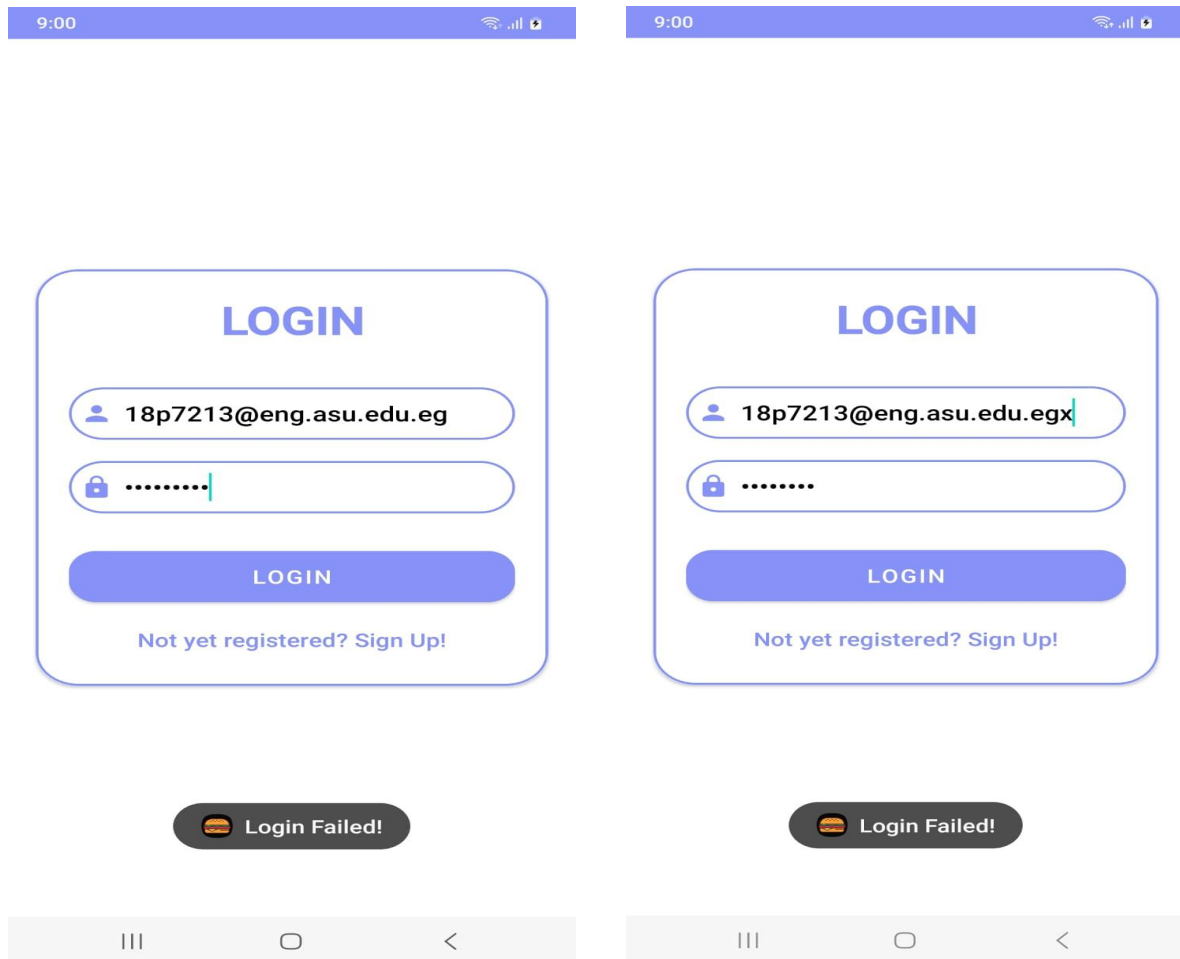
Only now after all required data is provided; The Sign Up operation is completed successfully:



Now lets check if the user we have just now created is found in the Firebase Authentication:



Now lets try to login but with a wrong password (a typo when writing the password which is “ainshams”), or try to login with the correct password, but with a typo in the email address; Login should fail in both cases and a Toast message should tell the user that the data he entered is not correct:

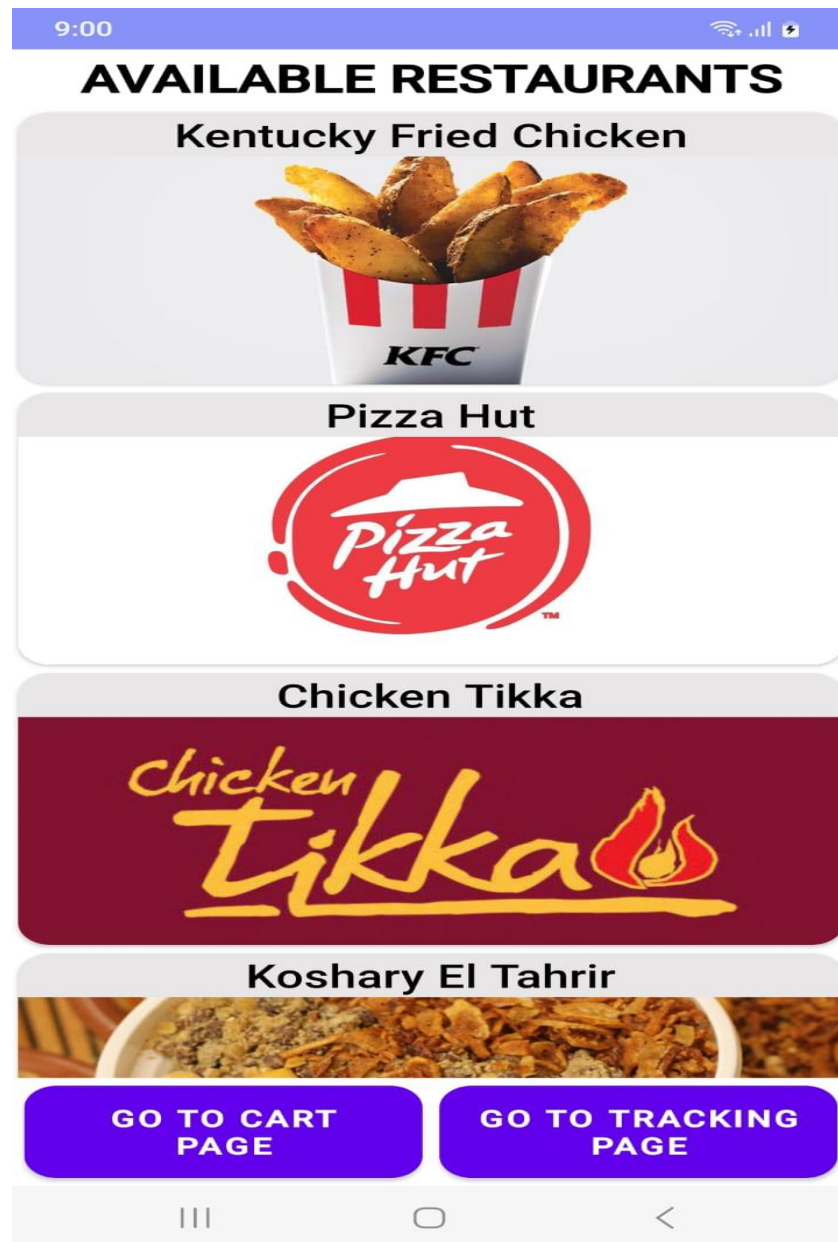


Now I will try to login with all information entered correctly, this will allow the user to enter the app to the Welcome Page where he is welcomed with his Full Name (retrieved from the ROOM DB), also there is a Toast message to the user to tell him that he logged in successfully:

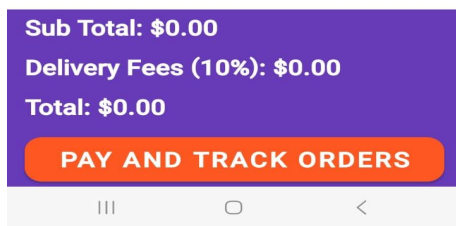


Now let's discover the app's main functionalities:

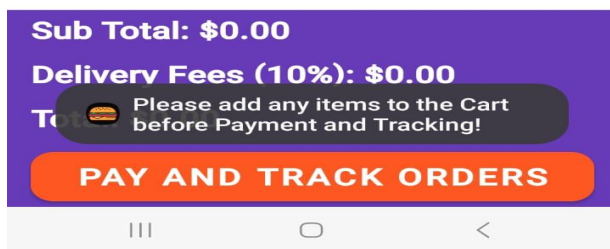
First let's check the page where all available restaurants are viewed to the user in a RecyclerView retrieved from the Realtime FB DB, this page is the page the user is redirected to once he presses on the "BROWSE RESTAURANTS" button found in the Welcome Page:



The user might press on the “GO TO CART PAGE” button or the “GO TO TRACKING PAGE” button to discover them before even adding any dish to the cart, they should both be completely empty initially of course:



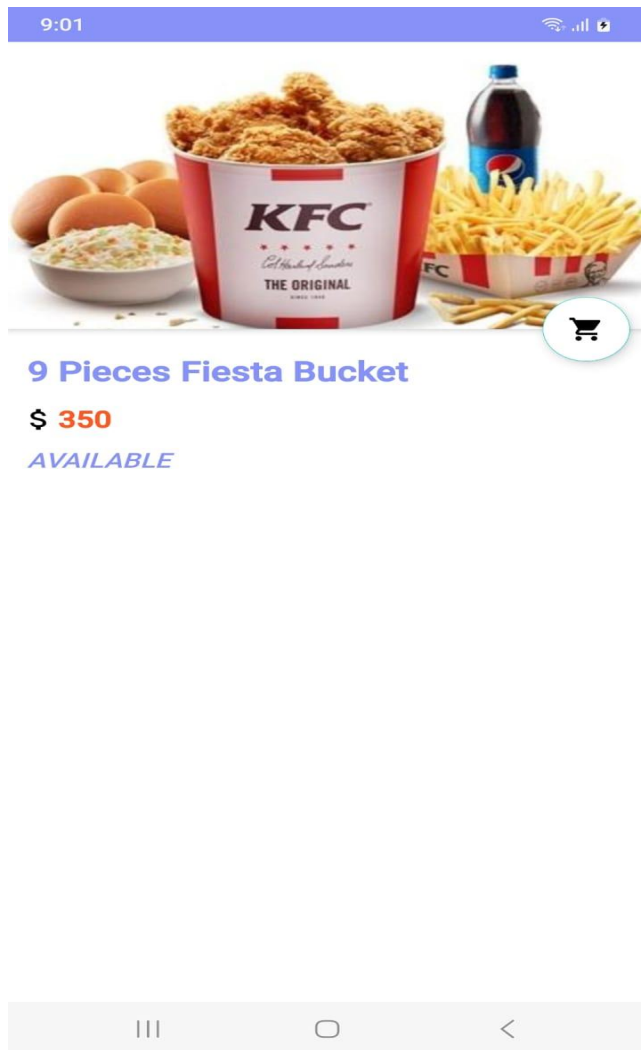
Now what if the user clicked in the Cart Page on the “PAY AND TRACK ORDERS” button while the cart is still empty? he should be told via a Toast message to firstly add anything to the cart before he can pay and start tracking his orders:



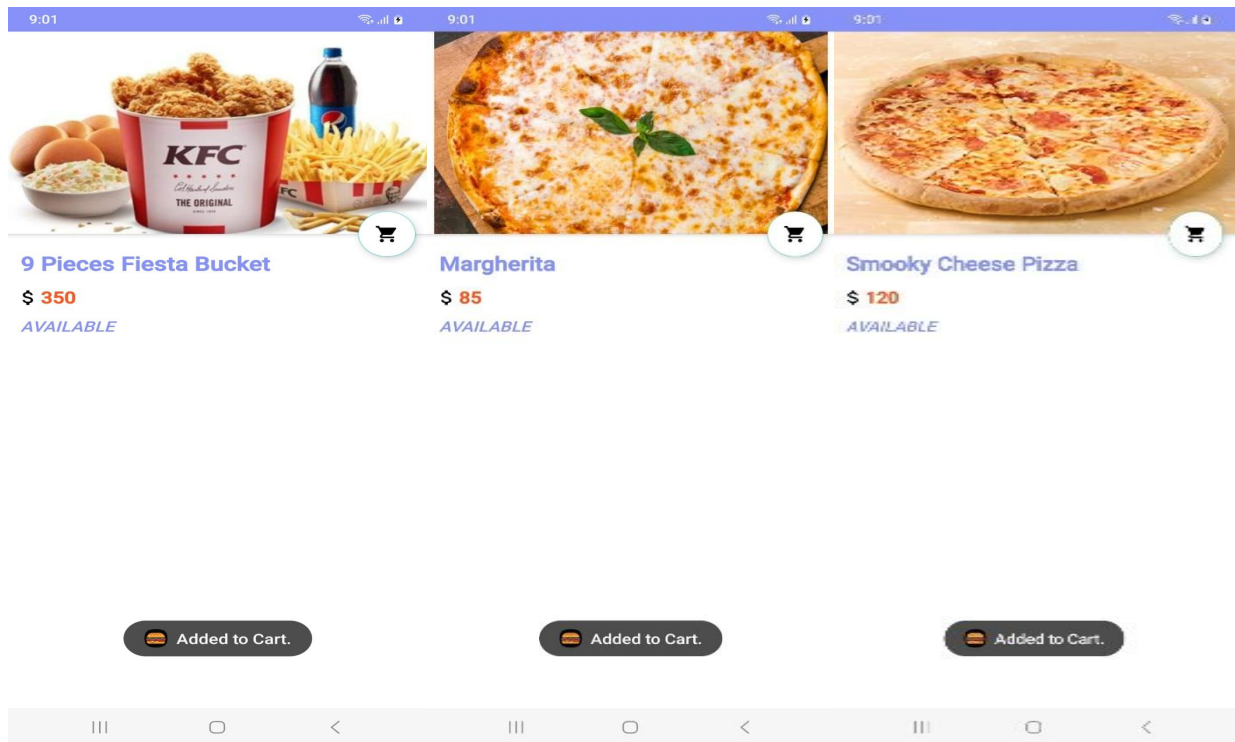
Now let's check the dishes of some restaurant, for example let's check Kentucky's dishes, they are also viewed in a RecyclerView and retrieved from the Realtime FB DB:



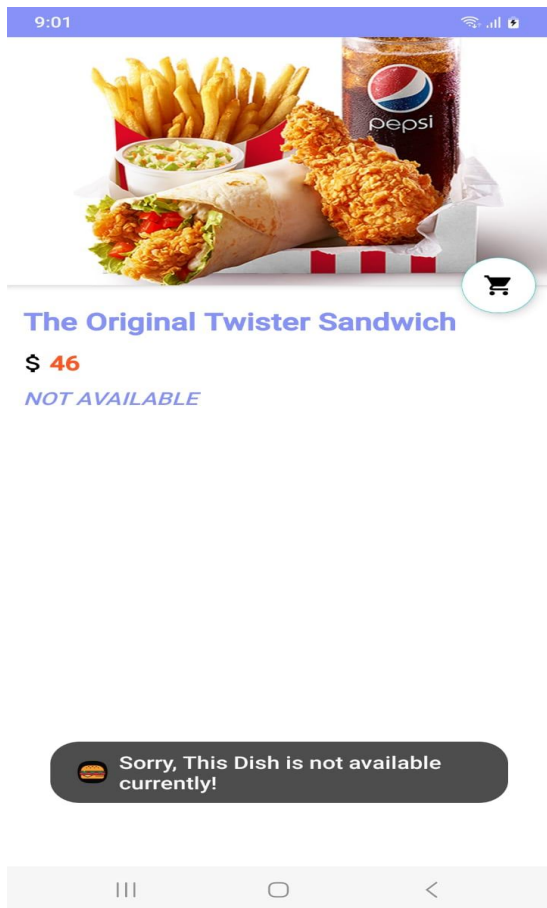
Now lets for example check the details of the “9 Pieces Fiesta Bucket” dish, details are; the dish image, the dish name, the dish price and the dish availability (all retrieved from the Realtime FB DB); also there is a Floating Button that has the icon of a Cart, to add a dish to the cart, the user should press on that button:



Now lets try to add some dishes to the Cart, for example I will add the “9 Pieces Fiesta Bucket” dish from KFC & the “Margherita” from Pizza Hut & 2 “Smoky Cheese Pizza” from Papa John’s (a dish is added to the cart more than once throughout pressing on the Cart Floating Button more than once). After any item is added to the cart, the user knows it through a Toast message:

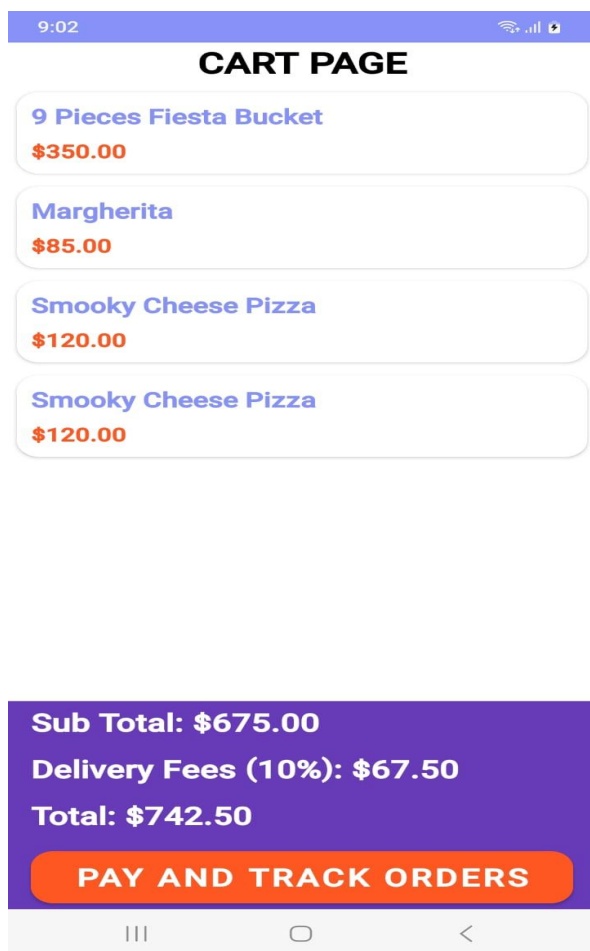


What if we tried to add a dish to the cart, but this dish is not available at the moment? A Toast message is viewed to the user to inform him that this dish is not available currently and of course the dish will not be added to the cart if the user tried to press the Cart Floating Button whatsoever:

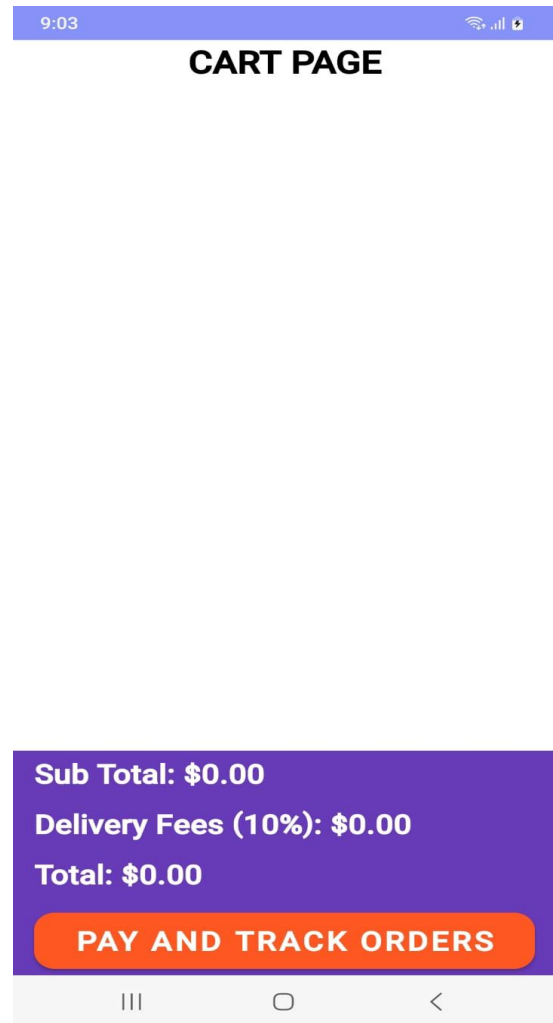
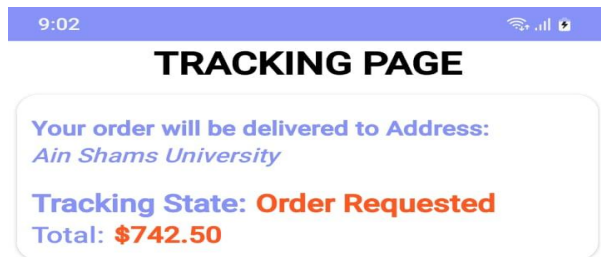


Now let's check if the cart got populated with what we added to it which is 1 dish from KFC & 1 dish from Pizza Hut & 2 dishes from Papa John's (both are the same dish requested twice by pressing the Cart Floating Button twice).

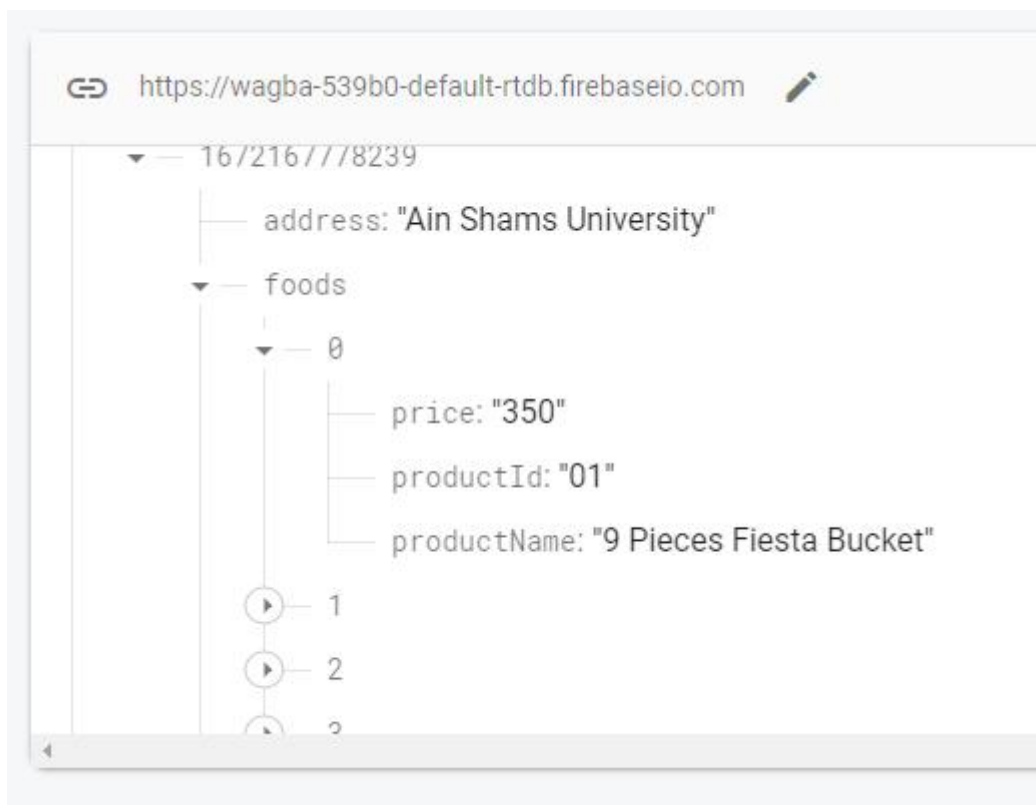
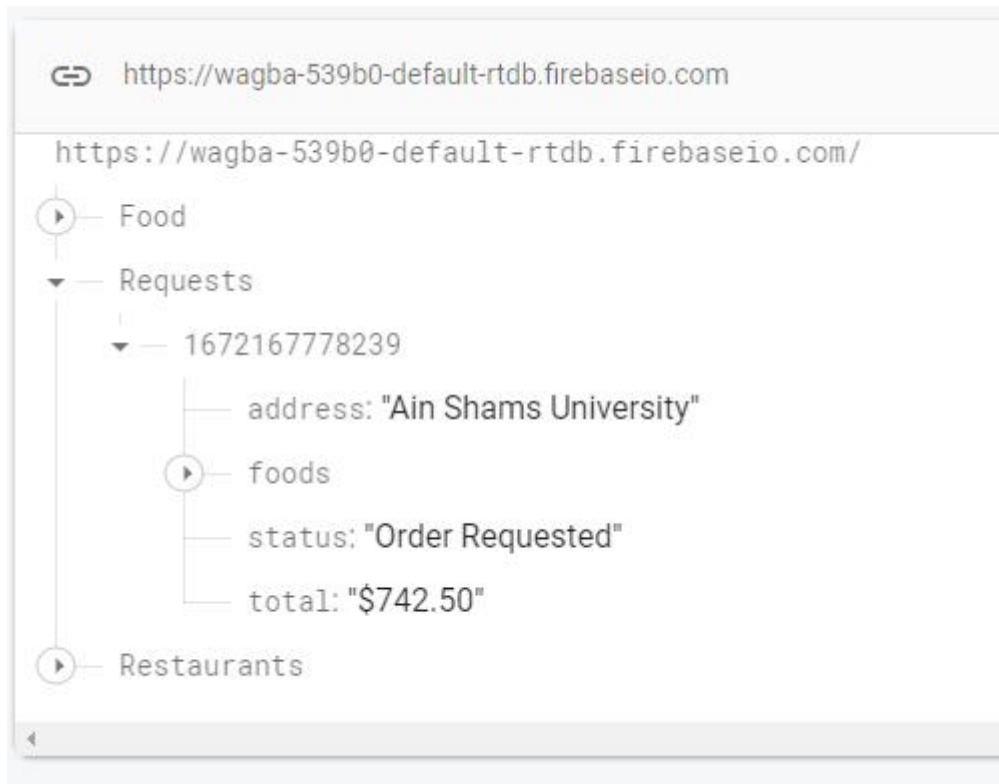
"Sub Total" is the sum of prices of all Cart Items; "Delivery Fees" are 10% of the "Sub Total" amount and "Total" is the summation of "Sub Total" and "Delivery Fees":



Now let's press on the "PAY AND TRACK ORDERS" button on the Cart page, when pressed, this button will clear the cart (this is clear in the 2nd picture of the current page) & send the order to the Tracking Page & inform the user of what just happened through a Toast message; of course the initial state of any order is that it is requested, also in the Tracking Page we retrieve the address from the ROOM DB:



Now let's check if this order was uploaded to the Realtime FB DB & let's also check if the 2nd app (named "WagbaHelper") received the order:



9:03



WagbaHelper

Order should be delivered to address:
Ain Shams University

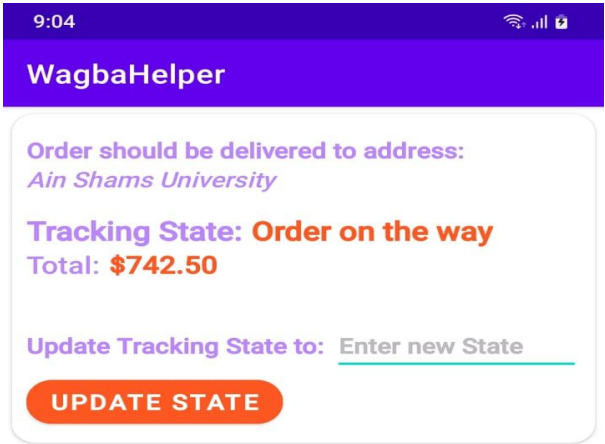
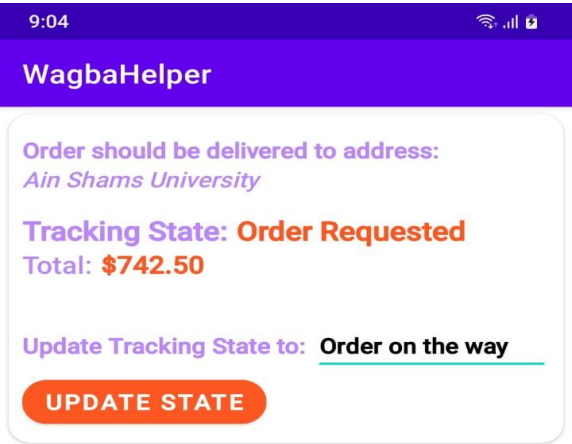
Tracking State: **Order Requested**
Total: **\$742.50**

Update Tracking State to:

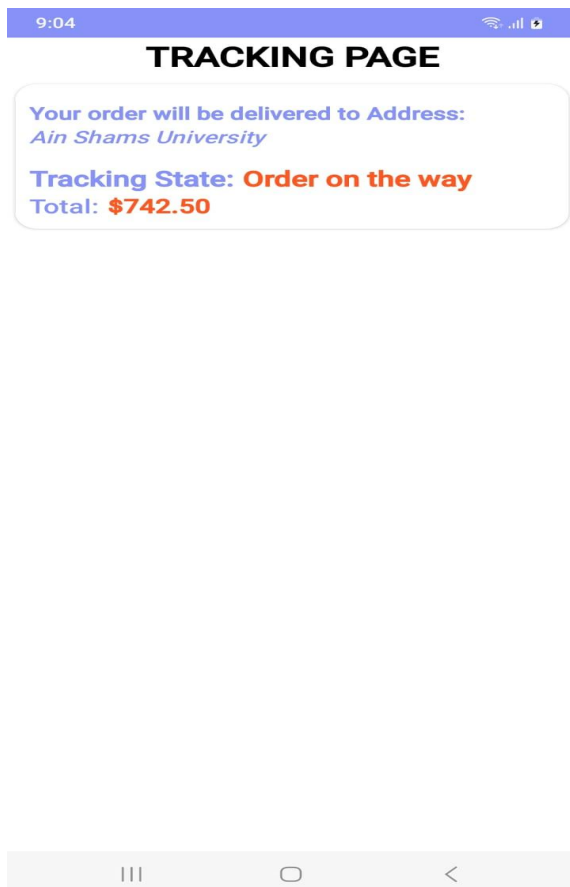
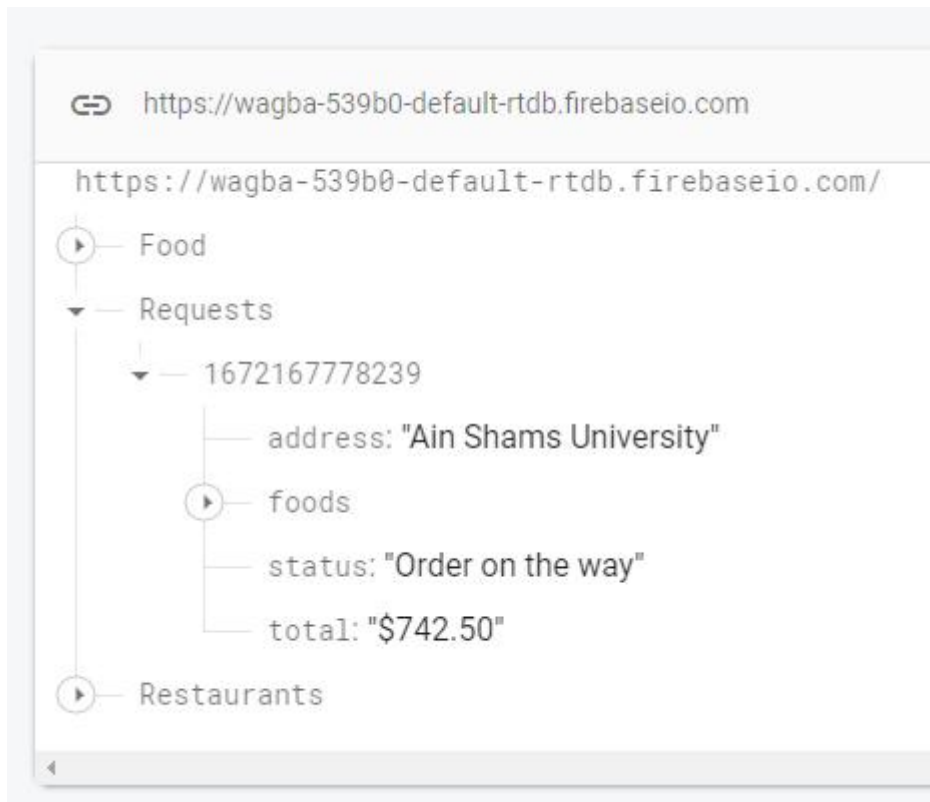
UPDATE STATE



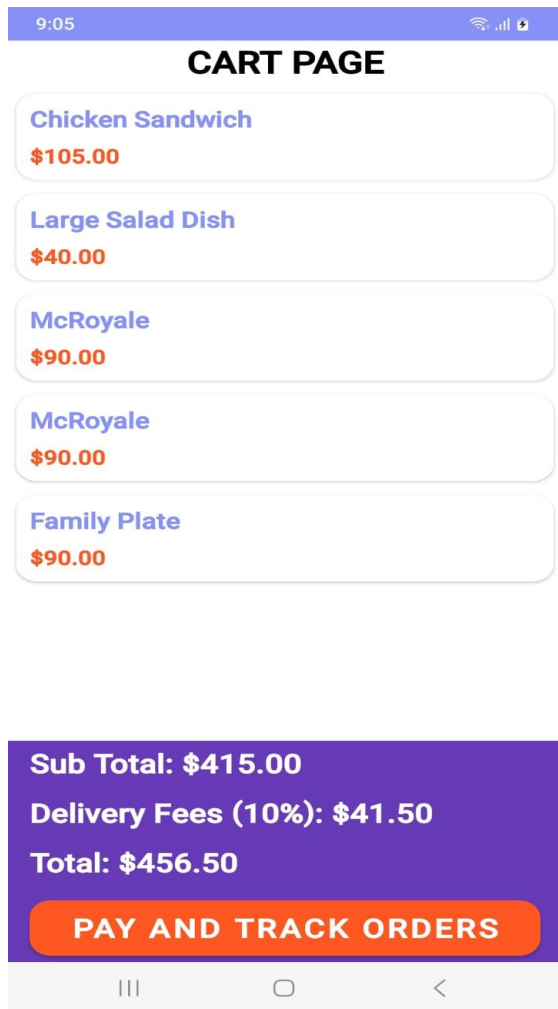
Now lets try to update the order’s tracking state from the “WagbaHelper” app:



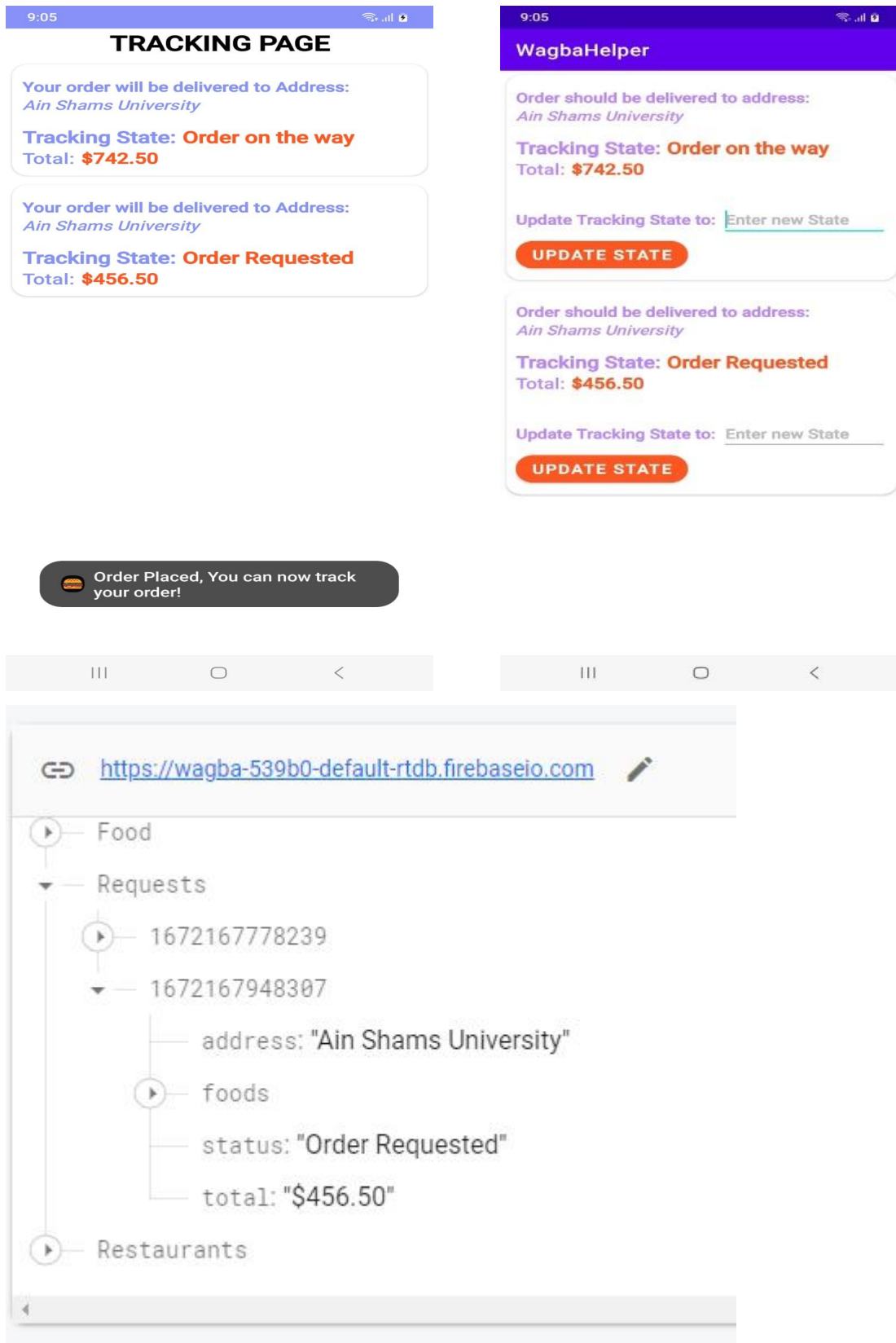
This state update from our 2nd helper app should be felt in both the FB DB and our main app:



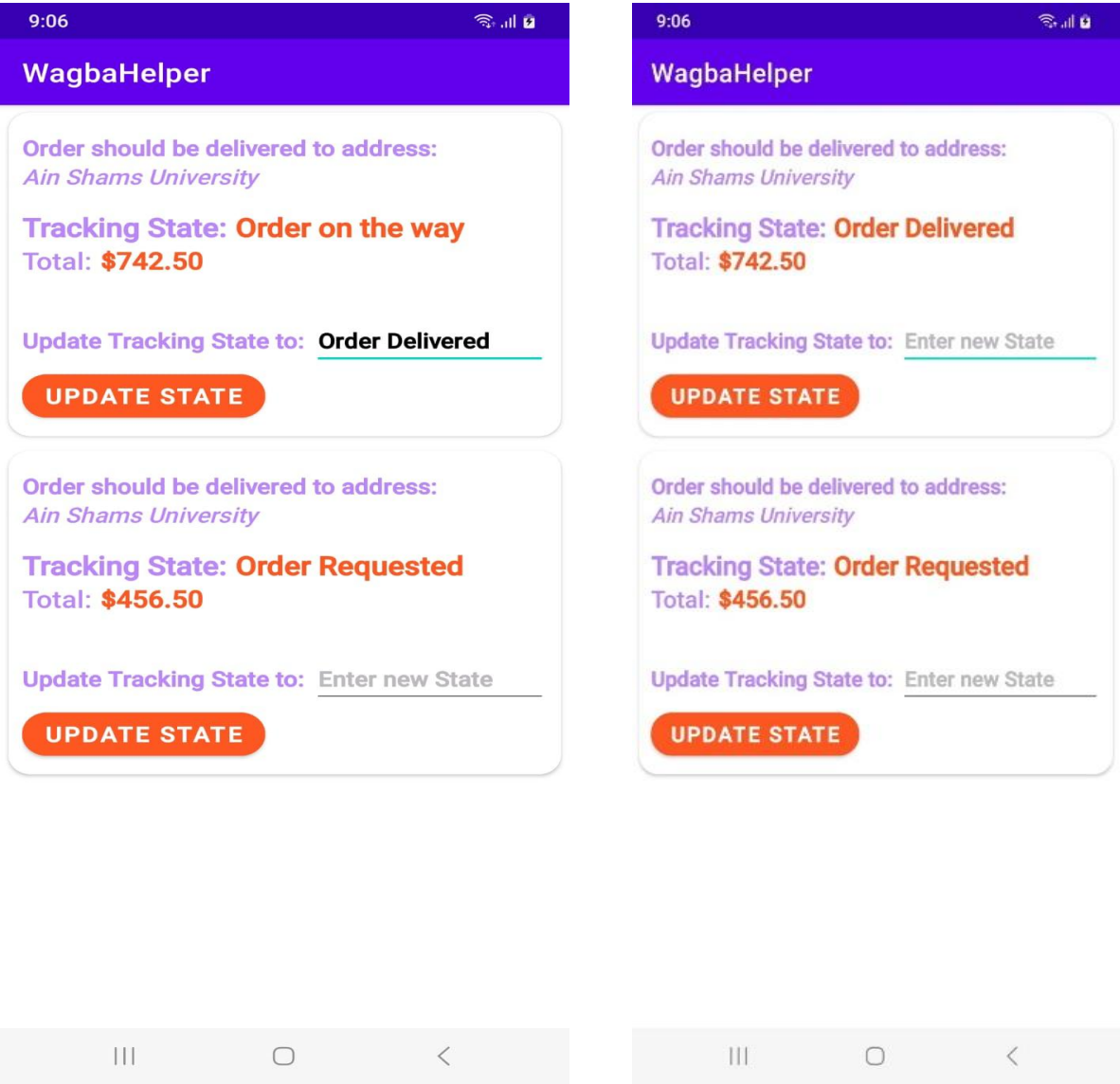
Now lets assume the user will, later on, order a 2nd order from the main app:



When the user presses on “PAY AND TRACK ORDERS” button, the same operation will repeat itself as shown in the coming screenshots:



Now lets further update the tracking state of both orders simultaneously to make sure everything is working as expected:



9:06

WagbaHelper

Order should be delivered to address:
Ain Shams University

Tracking State: **Order Delivered**
Total: **\$742.50**

Update Tracking State to:

UPDATE STATE

Order should be delivered to address:
Ain Shams University

Tracking State: **Order Requested**
Total: **\$456.50**

Update Tracking State to:

UPDATE STATE

|||

○

<

9:06

WagbaHelper

Order should be delivered to address:
Ain Shams University

Tracking State: **Order Delivered**
Total: **\$742.50**

Update Tracking State to:

UPDATE STATE

Order should be delivered to address:
Ain Shams University

Tracking State: **Order on the way**
Total: **\$456.50**

Update Tracking State to:

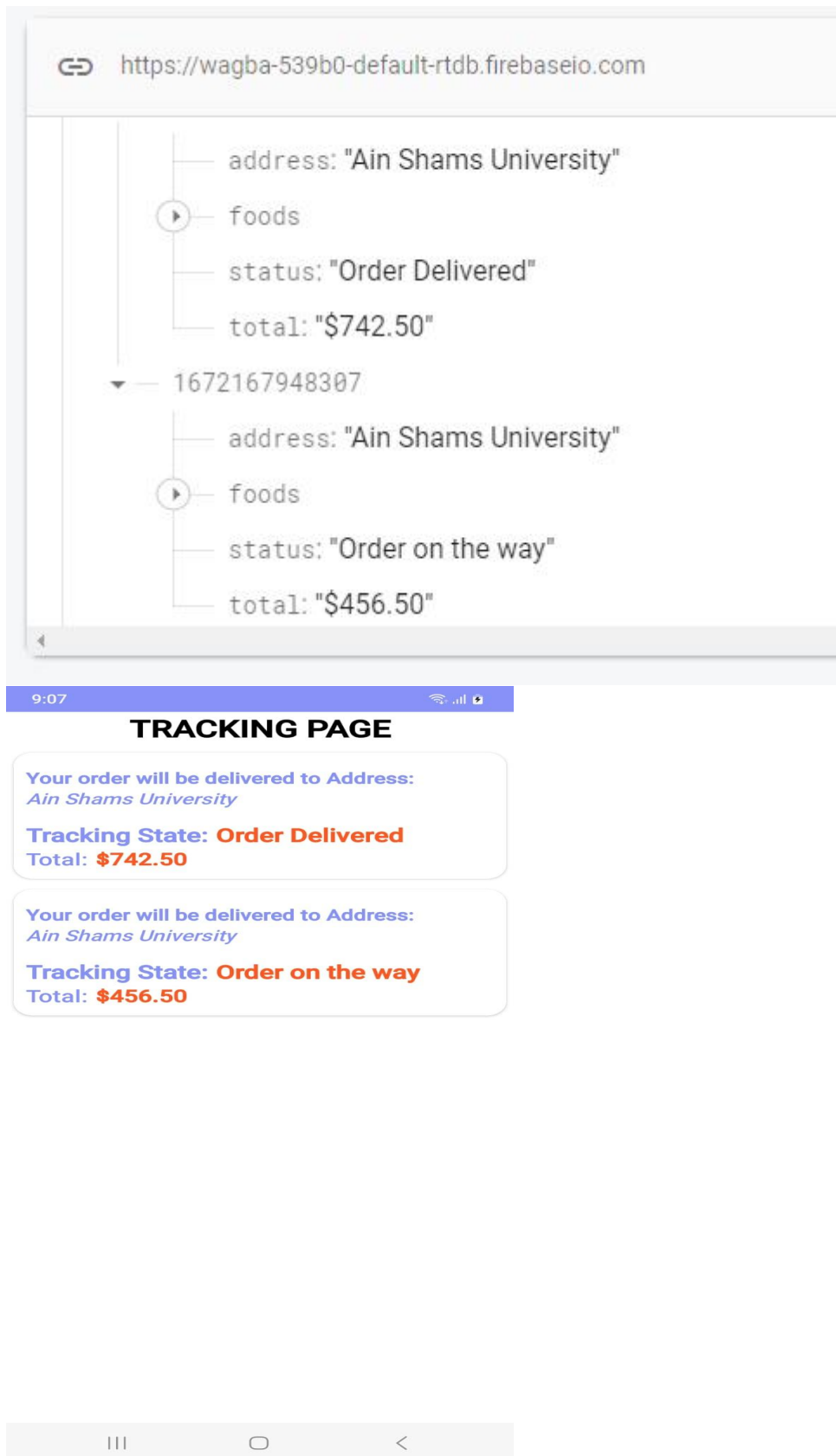
UPDATE STATE

|||

○

<

Lets check if our changes made a difference in the FB DB and the main app Tracking Page:



Now, lets discover both apps Java Code:

Main app's ("Wagba") Java Code:

MainActivity.java:

```
package com.example.wagba;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;

import com.example.wagba.restaurants.restaurantslist;

import java.util.List;

public class MainActivity extends AppCompatActivity {
    TextView fullName;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button button = findViewById(R.id.StartBTN);
        fullName = findViewById(R.id.userName);
        ProfileRoomDb profileRoomDb = ProfileRoomDb.getInstance(this);
        List<ProfileData> arrProfile = (List<ProfileData>)
profileRoomDb.profileDataDao().getAllData();
        String nameFromDB = arrProfile.get(0).getFullName();
        fullName.setText(nameFromDB);
        button.setOnClickListener(view -> startActivity(new
Intent(MainActivity.this, restaurantslist.class)));
    }
}
```

ItemClickListener.java (interface):

```
package com.example.wagba;
public interface ItemClickListener {
    void onClick(int position);
}
```

LoginActivity.java:

```
package com.example.wagba.authentication;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
```

```

import android.util.Patterns;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.example.wagba.MainActivity;
import com.example.wagba.R;
import com.google.firebase.auth.FirebaseAuth;

public class LoginActivity extends AppCompatActivity {
    private FirebaseAuth firebaseAuth;
    private EditText email, password;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        firebaseAuth = FirebaseAuth.getInstance();
        email = findViewById(R.id.LoginEmail);
        password = findViewById(R.id.LoginPassword);
        Button login = findViewById(R.id.LoginBTN);
        TextView signupRedirect = findViewById(R.id.SignUpRedirect);
        login.setOnClickListener(view -> {
            String mail = email.getText().toString();
            String pass = password.getText().toString();
            if(!mail.isEmpty() &&
Patterns.EMAIL_ADDRESS.matcher(mail).matches()) {
                if(!pass.isEmpty()) {
                    firebaseAuth.signInWithEmailAndPassword(mail, pass)
                        .addOnSuccessListener(authResult -> {
                            Toast.makeText(LoginActivity.this, "Login
Successful!", Toast.LENGTH_SHORT).show();
                            startActivity(new Intent(LoginActivity.this,
MainActivity.class));
                            finish();
                        }).addOnFailureListener(e ->
Toast.makeText(LoginActivity.this, "Login Failed!",
Toast.LENGTH_SHORT).show());
                } else
                    password.setError("Password can not be empty!");
            } else if(mail.isEmpty())
                email.setError("Email can not be empty!");
            else
                email.setError("Please enter a valid Email!");
        });
        signupRedirect.setOnClickListener(view -> startActivity(new
Intent(LoginActivity.this, SignUpActivity.class)));
    }
}

```

SignUpActivity.java:

```

package com.example.wagba.authentication;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;

```

```

import android.widget.TextView;
import android.widget.Toast;

import com.example.wagba.ProfileData;
import com.example.wagba.ProfileRoomDb;
import com.example.wagba.R;
import com.google.firebase.auth.FirebaseAuth;

import java.util.Objects;

public class SignUpActivity extends AppCompatActivity {
    private FirebaseAuth firebaseAuth;
    private EditText email, password, fullName, address;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sign_up);
        firebaseAuth = FirebaseAuth.getInstance();
        email = findViewById(R.id.SignUpEmail);
        password = findViewById(R.id.SignUpPassword);
        fullName = findViewById(R.id.SignUpName);
        address = findViewById(R.id.SignUpAddress);
        Button signUp = findViewById(R.id.SignUpBTN);
        TextView loginRedirect = findViewById(R.id.LoginRedirectTV);
        ProfileRoomDb profileRoomDb = ProfileRoomDb.getInstance(this);
        signUp.setOnClickListener(view -> {
            String user = email.getText().toString().trim();
            String pass = password.getText().toString().trim();
            String name = fullName.getText().toString();
            String addr = address.getText().toString();
            if(user.isEmpty())
                email.setError("Email can not be empty!");
            if(pass.isEmpty())
                password.setError("Password can not be empty!");
            if(name.isEmpty())
                fullName.setError("Please enter your Name!");
            if(addr.isEmpty())
                address.setError("Please enter your Address!");
            else{
                firebaseAuth.createUserWithEmailAndPassword(user,
pass).addOnCompleteListener(task -> {
                    if(task.isSuccessful()){
                        Toast.makeText(SignUpActivity.this, "Signed Up
Successfully!", Toast.LENGTH_SHORT).show();
                        profileRoomDb.profileDataDao().insert(new
ProfileData(name,addr));
                        startActivity(new Intent(SignUpActivity.this,
LoginActivity.class));
                    } else
                        Toast.makeText(SignUpActivity.this, "Signing Up
Failed!" + Objects.requireNonNull(task.getException()).getMessage(),
Toast.LENGTH_SHORT).show();
                });
            }
        });
        loginRedirect.setOnClickListener(view -> startActivity(new
Intent(SignUpActivity.this, LoginActivity.class)));
    }
}

```

Restaurant.java:

```
package com.example.wagba.restaurants;

public class Restaurant {
    String Name, Image;
    public Restaurant() {}
    public Restaurant(String name, String image) {
        Name = name;
        Image = image;
    }
    public String getName() {
        return Name;
    }
    public void setName(String name) {
        Name = name;
    }
    public String getImage() {
        return Image;
    }
    public void setImage(String image) {
        Image = image;
    }
}
```

MenuViewHolder.java:

```
package com.example.wagba.restaurants;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.example.wagba.ItemClickListener;
import com.example.wagba.R;

public class MenuViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener {
    public final TextView restaurantName;
    public final ImageView restaurantImage;
    private ItemClickListener itemClickListener;
    public MenuViewHolder(@NonNull View itemView) {
        super(itemView);
        restaurantName = (TextView)
itemView.findViewById(R.id.RestaurantNameTV);
        restaurantImage = (ImageView)
itemView.findViewById(R.id.RestaurantImage);
        itemView.setOnClickListener(this);
    }
    public void setItemClickListener(ItemClickListener itemClickListener) {
        this.itemClickListener = itemClickListener;
    }
    @Override
    public void onClick(View view) {
        itemClickListener.onClick(getBindingAdapterPosition());
    }
}
```

```
}  
}
```

RestaurantsList.java:

```
package com.example.wagba.restaurants;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.widget.Button;  
  
import com.example.wagba.R;  
import com.example.wagba.tracking.Tracking;  
import com.example.wagba.cart.Cart;  
import com.example.wagba.dishes.disheslist;  
import com.firebase.ui.database.FirebaseRecyclerAdapter;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.squareup.picasso.Picasso;  
  
public class restaurantslist extends AppCompatActivity {  
    FirebaseDatabase firebaseDatabase;  
    DatabaseReference databaseReference;  
    RecyclerView recyclerView;  
    RecyclerView.LayoutManager layoutManager;  
    FirebaseRecyclerAdapter<Restaurant, MenuViewHolder>  
    firebaseRecyclerAdapter;  
    Button cart, track;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_restaurantslist);  
        firebaseDatabase = FirebaseDatabase.getInstance();  
        databaseReference = firebaseDatabase.getReference("Restaurants");  
        recyclerView = (RecyclerView) findViewById(R.id.RestaurantsRV);  
        recyclerView.setHasFixedSize(true);  
        layoutManager = new LinearLayoutManager(this);  
        recyclerView.setLayoutManager(layoutManager);  
        loadRestaurants();  
        cart = (Button) findViewById(R.id.goToCartBTN);  
        track = (Button) findViewById(R.id.goToTrackingBTN);  
        cart.setOnClickListener(view -> {  
            Intent cartIntent = new Intent(restaurantslist.this,  
            Cart.class);  
            startActivity(cartIntent);  
        });  
        track.setOnClickListener(view -> {  
            Intent trackIntent = new Intent(restaurantslist.this,  
            Tracking.class);  
            startActivity(trackIntent);  
        });  
    }  
    private void loadRestaurants() {  
        firebaseRecyclerAdapter = new  
        FirebaseRecyclerAdapter<Restaurant,  
        MenuViewHolder>(Restaurant.class, R.layout.restaurantitem, MenuViewHolder.class,
```

```

ss, databaseReference) {
    @Override
    protected void populateViewHolder(MenuViewHolder viewHolder,
Restaurant model, int position) {
        viewHolder.restaurantName.setText(model.getName());

Picasso.with(getBaseContext()).load(model.getImage()).into(viewHolder.resta
urantImage);
        viewHolder.setItemClickListener((position1) -> {
            Intent dishesList = new Intent(restaurantslist.this,
disheslist.class);

dishesList.putExtra("RestaurantsID", firebaseRecyclerAdapter.getRef(position
1).getKey());
            startActivity(dishesList);
        });
    }
};
recyclerView.setAdapter(firebaseRecyclerAdapter);
}
}

```

Food.java:

```

package com.example.wagba.dishes;

public class Food {
    String Name, Price, Image, MenuID, Availability;
    public Food() {
    }
    public Food(String name, String price, String image, String menuID,
String availability) {
        Name = name;
        Price = price;
        Image = image;
        MenuID = menuID;
        Availability = availability;
    }
    public String getName() {
        return Name;
    }
    public void setName(String name) {
        Name = name;
    }
    public String getPrice() {
        return Price;
    }
    public String getImage() {
        return Image;
    }
    public String getAvailability() {
        return Availability;
    }
}

```

FoodViewHolder.java:

```
package com.example.wagba.dishes;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.example.wagba.ItemClickListener;
import com.example.wagba.R;

public class FoodViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener {
    public final TextView dishName;
    public final ImageView dishImage;
    private ItemClickListener itemClickListener;
    public void setItemClickListener(ItemClickListener itemClickListener) {
        this.itemClickListener = itemClickListener;
    }
    public FoodViewHolder(@NonNull View itemView) {
        super(itemView);
        dishName = (TextView) itemView.findViewById(R.id.DishNameTV);
        dishImage = (ImageView) itemView.findViewById(R.id.DishImage);
        itemView.setOnClickListener(this);
    }
    @Override
    public void onClick(View view) {
        itemClickListener.onClick(getBindingAdapterPosition());
    }
}
```

DishesList.java:

```
package com.example.wagba.dishes;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.content.Intent;
import android.os.Bundle;

import com.example.wagba.R;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.squareup.picasso.Picasso;

public class disheslist extends AppCompatActivity {
    RecyclerView recyclerView;
    RecyclerView.LayoutManager layoutManager;
    FirebaseDatabase firebaseDatabase;
    DatabaseReference databaseReference;
    String restaurantsID = "";
    FirebaseRecyclerAdapter<Food, FoodViewHolder> firebaseRecyclerAdapter;
    @Override
```



```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_disheslist);
    firebaseDatabase = FirebaseDatabase.getInstance();
    databaseReference = firebaseDatabase.getReference("Food");
    recyclerView = (RecyclerView) findViewById(R.id.DishesRV);
    recyclerView.setHasFixedSize(true);
    layoutManager = new LinearLayoutManager(this);
    recyclerView.setLayoutManager(layoutManager);
    if (getIntent() != null)
        restaurantsID = getIntent().getStringExtra("RestaurantsID");
    if (!restaurantsID.isEmpty())
        loadDishes(restaurantsID);
}

private void loadDishes(String restaurantsID) {
    firebaseRecyclerAdapter = new FirebaseRecyclerAdapter<Food,
FoodViewHolder>(Food.class, R.layout.dishitem, FoodViewHolder.class, databaseReference.orderByChild("MenuID").equalTo(restaurantsID)) {
        @Override
        protected void populateViewHolder(FoodViewHolder viewHolder,
Food model, int position) {
            viewHolder.dishName.setText(model.getName());

Picasso.with(getBaseContext()).load(model.getImage()).into(viewHolder.dishImage);

            viewHolder.setItemClickListener((position1) -> {
                Intent foodDetail = new Intent(disheslist.this,
FoodDetail.class);
                foodDetail.putExtra("FoodID",
firebaseRecyclerAdapter.getRef(position1).getKey());
                startActivity(foodDetail);
            });
        }
    };
    recyclerView.setAdapter(firebaseRecyclerAdapter);
}
}

```

FoodDetail.java:

```

package com.example.wagba.dishes;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.wagba.Database.Database;
import com.example.wagba.R;
import com.example.wagba.cart.Order;
import
com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

```

```

import com.google.firebase.database.ValueEventListener;
import com.squareup.picasso.Picasso;

public class FoodDetail extends AppCompatActivity {
    TextView food_name, food_price, food_availability;
    ImageView food_image;
    FloatingActionButton floatingActionButton;
    String foodId = "";
    FirebaseDatabase firebaseDatabase;
    DatabaseReference databaseReference;
    Food food;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_food_detail);
        food_availability = (TextView) findViewById(R.id.availability);
        firebaseDatabase = FirebaseDatabase.getInstance();
        databaseReference = firebaseDatabase.getReference("Food");
        floatingActionButton = (FloatingActionButton)
findViewById(R.id.CartBTN);
        floatingActionButton.setOnClickListener(view -> {
            if (food_availability.getText().equals("Available")) {
                new Database(getApplicationContext()).addToCart(new Order(
                    foodId,
                    food.getName(),
                    food.getPrice()
                ));
                Toast.makeText(FoodDetail.this, "Added to Cart.",
Toast.LENGTH_SHORT).show();
            }
            else {
                Toast.makeText(FoodDetail.this, "Sorry, This Dish is not
available currently!", Toast.LENGTH_SHORT).show();
            }
        });
        food_name = (TextView) findViewById(R.id.foodName);
        food_price = (TextView) findViewById(R.id.foodPrice);
        food_image = (ImageView) findViewById(R.id.foodImage);
        if(getIntent() != null)
            foodId = getIntent().getStringExtra("FoodID");
        if(!foodId.isEmpty()){
            getFoodDetail(foodId);
        }
    }
    private void getFoodDetail(String foodId) {
        databaseReference.child(foodId).addValueEventListener(new
ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                food = dataSnapshot.getValue(Food.class);
                Picasso.with(getApplicationContext()).load(food.getImage()).into(food_image);
                food_price.setText(food.getPrice());
                food_name.setText(food.getName());
                food_availability.setText(food.getAvailability());
            }
            @Override
            public void onCancelled(@NonNull DatabaseError error) {
            }
        });
    }
}

```

```
}  
}
```

Database.java:

```
package com.example.wagba.Database;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteQueryBuilder;  
  
import com.example.wagba.cart.Order;  
import com.readystatesoftware.sqliteasset.SQLiteAssetHelper;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class Database extends SQLiteAssetHelper {  
    private static final String DB_NAME = "WagbaDB.db";  
    private static final int DB_VER=1;  
    public Database(Context context) {  
        super(context, DB_NAME,null, DB_VER);  
    }  
    public List<Order> getCart() {  
        SQLiteDatabase sqLiteDatabase = getReadableDatabase();  
        SQLiteQueryBuilder sqLiteQueryBuilder = new SQLiteQueryBuilder();  
        String[] sqlSelect = {"ProductId", "ProductName", "Price"};  
        String sqlTable = "OrderDetail";  
        sqLiteQueryBuilder.setTables(sqlTable);  
        Cursor cursor =  
sqLiteQueryBuilder.query(sqLiteDatabase, sqlSelect, null, null, null, null, null);  
        final List<Order> result = new ArrayList<>();  
        if(cursor.moveToFirst()) {  
            do{  
                result.add(new  
Order(cursor.getString(cursor.getColumnIndex("ProductId")),  
  
cursor.getString(cursor.getColumnIndex("ProductName")),  
                cursor.getString(cursor.getColumnIndex("Price"))  
                ));  
            }while(cursor.moveToNext());  
        }  
        return result;  
    }  
    public void addToCart(Order order){  
        SQLiteDatabase sqLiteDatabase = getReadableDatabase();  
        String query = String.format("INSERT INTO  
OrderDetail(ProductId,ProductName,Price) VALUES ('%s','%s','%s');",  
                order.getId(),  
                order.getProductName(),  
                order.getPrice());  
        sqLiteDatabase.execSQL(query);  
    }  
    public void cleanCart(){  
        SQLiteDatabase sqLiteDatabase = getReadableDatabase();  
        String query = "DELETE FROM OrderDetail";  
        sqLiteDatabase.execSQL(query);  
    }  
}
```

Order.java:

```
package com.example.wagba.cart;

public class Order {
    private String ProductId;
    private String ProductName;
    private String Price;
    public Order() {
    }
    public Order(String productId, String productName, String price) {
        ProductId = productId;
        ProductName = productName;
        Price = price;
    }
    public String getProductId() {
        return ProductId;
    }
    public String getProductName() {
        return ProductName;
    }
    public String getPrice() {
        return Price;
    }
}
```

CartAdapter & CartViewHolder .java:

```
package com.example.wagba.cart;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.example.wagba.R;

import java.text.NumberFormat;
import java.util.List;
import java.util.Locale;

class CartViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener{
    public final TextView name;
    public final TextView price;
    public CartViewHolder(@NonNull View itemView) {
        super(itemView);
        name = (TextView) itemView.findViewById(R.id.cartItem);
        price = (TextView) itemView.findViewById(R.id.cartItemPrice);
    }
    @Override
    public void onClick(View view) {
    }
}
```

```

}
public class CartAdapter extends RecyclerView.Adapter<CartViewHolder> {
    private final List<Order> listData;
    private final Context context;
    public CartAdapter(List<Order> listData, Context context) {
        this.listData = listData;
        this.context = context;
    }
    @NonNull
    @Override
    public CartViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        LayoutInflater inflater = LayoutInflater.from(context);
        View view =
inflater.inflate(R.layout.cart_layout, parent, false);
        return new CartViewHolder(view);
    }
    @Override
    public void onBindViewHolder(@NonNull CartViewHolder holder, int
position) {
        Locale locale = new Locale("en", "US");
        NumberFormat numberFormat =
NumberFormat.getCurrencyInstance(locale);
        int Price = (Integer.parseInt(listData.get(position).getPrice()));
        holder.price.setText(numberFormat.format(Price));
        holder.name.setText(listData.get(position).getProductName());
    }
    @Override
    public int getItemCount() {
        return listData.size();
    }
}

```

Cart.java:

```

package com.example.wagba.cart;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import com.example.wagba.Database.Database;
import com.example.wagba.ProfileData;
import com.example.wagba.ProfileRoomDb;
import com.example.wagba.R;
import com.example.wagba.tracking.Request;
import com.example.wagba.tracking.Tracking;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

import java.text.NumberFormat;
import java.util.ArrayList;
import java.util.List;
import java.util.Locale;

```

```

public class Cart extends AppCompatActivity {
    RecyclerView recyclerView;
    RecyclerView.LayoutManager layoutManager;
    FirebaseDatabase firebaseDatabase;
    TextView subT, deliveryT, T;
    Button button;
    List<Order> cart = new ArrayList<>();
    CartAdapter cartAdapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_cart);
        firebaseDatabase = FirebaseDatabase.getInstance();
        recyclerView = (RecyclerView) findViewById(R.id.listCart);
        recyclerView.setHasFixedSize(true);
        layoutManager = new LinearLayoutManager(this);
        recyclerView.setLayoutManager(layoutManager);
        subT = (TextView) findViewById(R.id.subTotal);
        deliveryT = (TextView) findViewById(R.id.deliveryTotal);
        T = (TextView) findViewById(R.id.total);
        button = (Button) findViewById(R.id.btnPlaceOrder);
        ProfileRoomDb profileRoomDb = ProfileRoomDb.getInstance(this);
        List<ProfileData> arrProfile = (List<ProfileData>)
profileRoomDb.profileDataDao().getAllData();
        String addressFromDB = arrProfile.get(0).getAddress();
        DatabaseReference databaseReference;
        databaseReference = firebaseDatabase.getReference("Requests");
        String status = "Order Requested";
        button.setOnClickListener(view -> {
            if(cartAdapter.getItemCount()==0)
                Toast.makeText(Cart.this,"Please add any items to the Cart
before Payment and Tracking!",Toast.LENGTH_LONG).show();
            else {
                Request request = new Request(status, addressFromDB,
T.getText().toString(), cart);

databaseReference.child(String.valueOf(System.currentTimeMillis())).setValu
e(request);

                new Database(getBaseContext()).cleanCart();
                Toast.makeText(Cart.this, "Order Placed, You can now track
your order!", Toast.LENGTH_SHORT).show();
                startActivity(new Intent(Cart.this, Tracking.class));
                finish();
            }
        });
        loadCart();
    }
    private void loadCart() {
        cart = new Database(this).getCarts();
        cartAdapter = new CartAdapter(cart,this);
        recyclerView.setAdapter(cartAdapter);
        int SubTotal = 0;
        for(Order order:cart)
            SubTotal+=(Integer.parseInt(order.getPrice()));
        Locale locale = new Locale("en","US");
        NumberFormat numberFormat =
NumberFormat.getCurrencyInstance(locale);
        subT.setText(numberFormat.format(SubTotal));
        double DeliveryTotal = SubTotal*0.1;
        double Total = SubTotal+DeliveryTotal;
    }
}

```

```

        deliveryT.setText(numberFormat.format(DeliveryTotal));
        T.setText(numberFormat.format(Total));
    }
}

```

ProfileData.java:

```

package com.example.wagba;
import androidx.room.ColumnInfo;
import androidx.room.Entity;
import androidx.room.Ignore;
import androidx.room.PrimaryKey;

@Entity(tableName = "ProfileData")
public class ProfileData {
    @PrimaryKey(autoGenerate = true)
    private int id;
    @ColumnInfo(name = "fullName")
    private final String fullName;
    @ColumnInfo(name = "address")
    private final String address;
    public ProfileData(int id, String fullName, String address) {
        this.fullName = fullName;
        this.address = address;
        this.id = id;
    }
    @Ignore
    public ProfileData(String fullName, String address) {
        this.fullName = fullName;
        this.address = address;
    }
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getFullName() {
        return fullName;
    }
    public String getAddress() {
        return address;
    }
}

```

ProfileDataDao.java (interface):

```

package com.example.wagba;
import androidx.room.Dao;
import androidx.room.Insert;
import androidx.room.Query;

import java.util.List;

@Dao
public interface ProfileDataDao {
    @Insert

```

```

    void insert(ProfileData profileData);
    @Query("SELECT * FROM ProfileData")
    List<ProfileData> getAllData();
}

```

ProfileRepository.java:

```

package com.example.wagba;
import android.app.Application;
import android.os.AsyncTask;

import java.util.List;

public class ProfileRepository {
    private final ProfileDataDao mProfileDao;
    private final List<ProfileData> getAllData;
    public ProfileRepository(Application app){
        ProfileRoomDb db = ProfileRoomDb.getInstance(app);
        mProfileDao = db.profileDataDao();
        getAllData = mProfileDao.getAllData();
    }
    public void insert(ProfileData profileData) {
        new InsertAsyncTask(mProfileDao).execute(profileData);
    }
    public List<ProfileData> getAllData(){
        return getAllData;
    }
    private static class InsertAsyncTask extends
    AsyncTask<ProfileData,Void,Void>{
        private final ProfileDataDao mProfileDao;
        public InsertAsyncTask(ProfileDataDao profileDataDao){
            mProfileDao = profileDataDao;
        }
        @Override
        protected Void doInBackground(ProfileData... profileData) {
            mProfileDao.insert(profileData[0]);
            return null;
        }
    }
}

```

ProfileRoomDb.java:

```

package com.example.wagba;
import android.content.Context;
import android.os.AsyncTask;

import androidx.annotation.NonNull;
import androidx.room.Database;
import androidx.room.Room;
import androidx.room.RoomDatabase;
import androidx.sqlite.db.SupportSQLiteDatabase;

@Database(entities = ProfileData.class, exportSchema = false, version = 1)
public abstract class ProfileRoomDb extends RoomDatabase {
    private static ProfileRoomDb instance;
    public abstract ProfileDataDao profileDataDao();
}

```



```

        public static synchronized ProfileRoomDb getInstance(Context context){
            if(instance == null){
                instance = Room.databaseBuilder(context.getApplicationContext(),
ProfileRoomDb.class, "profile-
database").fallbackToDestructiveMigration().allowMainThreadQueries().build()
;
            }
            return instance;
        }
        private static final RoomDatabase.Callback roomCallBack = new
RoomDatabase.Callback() {
            @Override
            public void onCreate(@NonNull SupportSQLiteDatabase db) {
                super.onCreate(db);
                new PopulateDataAsyncTask(instance).execute();
            }
            @Override
            public void onOpen(@NonNull SupportSQLiteDatabase db) {
                super.onOpen(db);
            }
        };
        private static class PopulateDataAsyncTask extends
AsyncTask<Void,Void,Void>{
            PopulateDataAsyncTask(ProfileRoomDb db){
                ProfileDataDao mProfileDao = db.profileDataDao();
            }
            @Override
            protected Void doInBackground(Void... voids) {
                return null;
            }
        }
    }
}

```

Request.java:

```

package com.example.wagba.tracking;
import com.example.wagba.cart.Order;

import java.util.List;

public class Request {
    private String status;
    private String address;
    private String total;
    private List<Order> foods;
    public Request() {
    }
    public Request(String status, String address, String total, List<Order>
foods) {
        this.status = status;
        this.address = address;
        this.total = total;
        this.foods = foods;
    }
    public String getStatus() {
        return status;
    }
    public void setStatus(String status) {
        this.status = status;
    }
}

```

```

    }
    public String getAddress() {
        return address;
    }
    public void setAddress(String address) {
        this.address = address;
    }
    public String getTotal() {
        return total;
    }
    public void setTotal(String total) {
        this.total = total;
    }
    public List<Order> getFoods() {
        return foods;
    }
    public void setFoods(List<Order> foods) {
        this.foods = foods;
    }
}

```

Track.java:

```

package com.example.wagba.tracking;

public class Track {
    String address, total, status;
    public Track() {
    }
    public Track(String address, String total, String status) {
        this.address = address;
        this.total = total;
        this.status = status;
    }
    public String getAddress() {
        return address;
    }
    public void setAddress(String address) {
        this.address = address;
    }
    public String getTotal() {
        return total;
    }
    public void setTotal(String total) {
        this.total = total;
    }
    public String getStatus() {
        return status;
    }
    public void setStatus(String status) {
        this.status = status;
    }
}

```

TrackViewHolder.java:

```

package com.example.wagba.tracking;

```

```

import android.view.View;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.example.wagba.R;

public class TrackViewHolder extends RecyclerView.ViewHolder {
    public final TextView address;
    public final TextView total;
    public final TextView status;
    public TrackViewHolder(@NonNull View itemView) {
        super(itemView);
        address = (TextView) itemView.findViewById(R.id.addressTracking);
        total = (TextView) itemView.findViewById(R.id.totalTracking);
        status = (TextView) itemView.findViewById(R.id.statusTracking);
    }
}

```

Tracking.java:

```

package com.example.wagba.tracking;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.os.Bundle;

import com.example.wagba.R;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

public class Tracking extends AppCompatActivity {
    FirebaseDatabase firebaseDatabase;
    DatabaseReference databaseReference;
    RecyclerView recyclerView;
    RecyclerView.LayoutManager layoutManager;
    FirebaseRecyclerAdapter<Track, TrackViewHolder> firebaseRecyclerAdapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tracking);
        firebaseDatabase = FirebaseDatabase.getInstance();
        databaseReference = firebaseDatabase.getReference("Requests");
        recyclerView = (RecyclerView) findViewById(R.id.listTracking);
        recyclerView.setHasFixedSize(true);
        layoutManager = new LinearLayoutManager(this);
        recyclerView.setLayoutManager(layoutManager);
        loadRequests();
    }
    private void loadRequests() {
        firebaseRecyclerAdapter = new FirebaseRecyclerAdapter<Track,
TrackViewHolder>(Track.class, R.layout.tracking_item, TrackViewHolder.class, d
atabaseReference) {
            @Override
            protected void populateViewHolder(TrackViewHolder viewHolder,
Track model, int position) {

```

```

        viewHolder.address.setText(model.getAddress());
        viewHolder.total.setText(model.getTotal());
        viewHolder.status.setText(model.getStatus());
    }
};
recyclerView.setAdapter(firebaseRecyclerAdapter);
}
}

```

Helper app's ("WagbaHelper") Java Code:

MainActivity.java:

```

package com.example.wagbahelper;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.os.Bundle;
import android.view.View;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

import java.util.HashMap;
import java.util.Map;

public class MainActivity extends AppCompatActivity {
    FirebaseDatabase firebaseDatabase;
    DatabaseReference databaseReference;
    RecyclerView recyclerView;
}

```

```

RecyclerView.LayoutManager layoutManager;
FirebaseRecyclerAdapter<Track,TrackViewHolder> firebaseRecyclerAdapter;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    firebaseDatabase = FirebaseDatabase.getInstance();
    databaseReference = firebaseDatabase.getReference("Requests");
    recyclerView = (RecyclerView) findViewById(R.id.listTracking);
    recyclerView.setHasFixedSize(true);
    layoutManager = new LinearLayoutManager(this);
    recyclerView.setLayoutManager(layoutManager);
    loadRequests();
}
private void loadRequests() {
    firebaseRecyclerAdapter = new FirebaseRecyclerAdapter<Track,
TrackViewHolder>(Track.class,R.layout.tracking_item,TrackViewHolder.class,d
atabaseReference) {
        @Override
        protected void populateViewHolder(TrackViewHolder viewHolder,
Track model, int position) {
            viewHolder.address.setText(model.getAddress());
            viewHolder.total.setText(model.getTotal());
            viewHolder.status.setText(model.getStatus());
            viewHolder.button.setOnClickListener(new
View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    String newState =
viewHolder.editText.getText().toString();
                    HashMap hashMap = new HashMap();
                    hashMap.put("status",newState);

                    databaseReference.child(getRef(position).getKey()).updateChildren(hashMap);
                    viewHolder.editText.setText("");
                }
            });
        }
    };
    recyclerView.setAdapter(firebaseRecyclerAdapter);
}
}

```

Track.java:

```

package com.example.wagbahelper;
public class Track {
    String address, total, status;
    public Track() {
    }
    public Track(String address, String total, String status) {
        this.address = address;
        this.total = total;
        this.status = status;
    }
}

```

```

    }
    public String getAddress() {
        return address;
    }
    public void setAddress(String address) {
        this.address = address;
    }
    public String getTotal() {
        return total;
    }
    public void setTotal(String total) {
        this.total = total;
    }
    public String getStatus() {
        return status;
    }
    public void setStatus(String status) {
        this.status = status;
    }
}

```

TrackViewHolder.java:

```

package com.example.wagbahelper;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

public class TrackViewHolder extends RecyclerView.ViewHolder {
    public final TextView address;
    public final TextView total;
    public final TextView status;
    public EditText editText;
    public Button button;
    public TrackViewHolder(@NonNull View itemView) {
        super(itemView);
        address = (TextView) itemView.findViewById(R.id.addressTracking);
        total = (TextView) itemView.findViewById(R.id.totalTracking);
        status = (TextView) itemView.findViewById(R.id.statusTracking);
        editText = (EditText) itemView.findViewById(R.id.newState);
        button = (Button) itemView.findViewById(R.id.updateStateBTN);
    }
}

```