Abstract

Sabzi Mart is Ecommerce Application. It is store for buying vegetables. The idea come to me during pandemic break The Service is for the people near the official stores. This project is of great importance if few more features added like pay via jazzcash and palpay the idea can be successful. The main goal of the project was to provide the user service of common food like vegetable at fair price and of good qualities .application consists of two main part. One for the admin and one for the customer. Admin handle events like add product and see new orders being placed. Customer can buy products of his needs. Firebase database is used to both entertain admin and customers.

1. Introduction

We all are aware of life being affected by this pandemic. During this Pandemic break. Price of several product was increased or sale at high rate by the retailers. Customer added no choice other than to buy it from retailer. Me myself had to go to main vegetable mart once a week to buy it at fair rate at early in the morning. The struggle was for everyone. The Main stores like metro product was sale at high rate and quality was also compromised. My product is Sabzi Mart.

As the name describe it is Store to buy vegetable and delivery to home. The service is for the people near the store or within 10-15km. The main Motivation of the project was to resolve the food issue arise during pandemic and for middle to upper class family to buy food of good quality and fair price at their doorsteps. Every service is getting online day by day. Introduction of this idea can change the world especially for the aboard countries where coronavirus is getting faille day by day.

Since all the facts exit in real world if handle correctly then this idea can get great response. For knowledge based this project was quite helpful for me learning the flow of ecommerce app and how it is implemented on step by step.

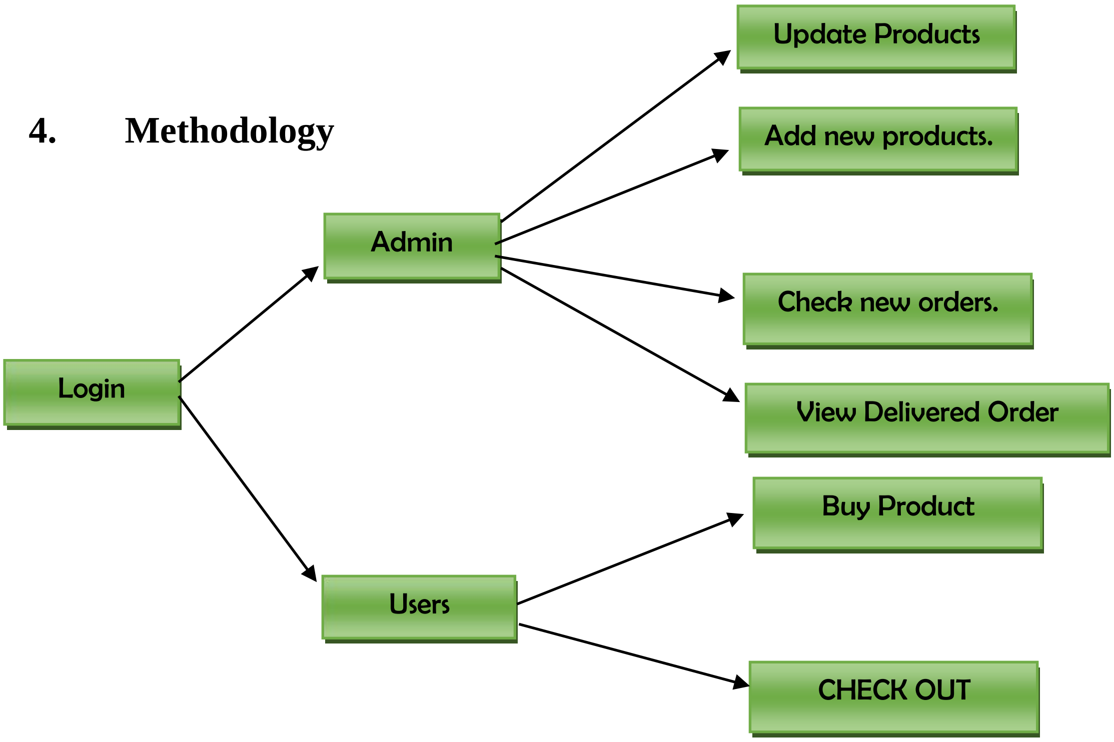
2. Problem Statement

- Pandemic Makes the food shopping difficult people are buying at high rates and some are comprising on high rates.
- Buying sale for higher ups or upper class was done by their servants during this pandemic. This way is dangerous.

3. Objectives

- Make the food shopping easy for everyone.
- Quality food at fair price at your doorsteps
- Reduce the Chances of coronavirus

4. Methodology



Flow of the Product is below

- Main Activity asking the user to login or signup.
- If Signup Sign up activity phone, name and password would ask for the user and Store to the firebase real time database under users Child.
- If login then Login Activity would ask the users to enter phone and password and chose as a user or as admin
- If on choice the phone and password are checked from firebase Realtime data and on successfully according the category new activity is open

If login as users

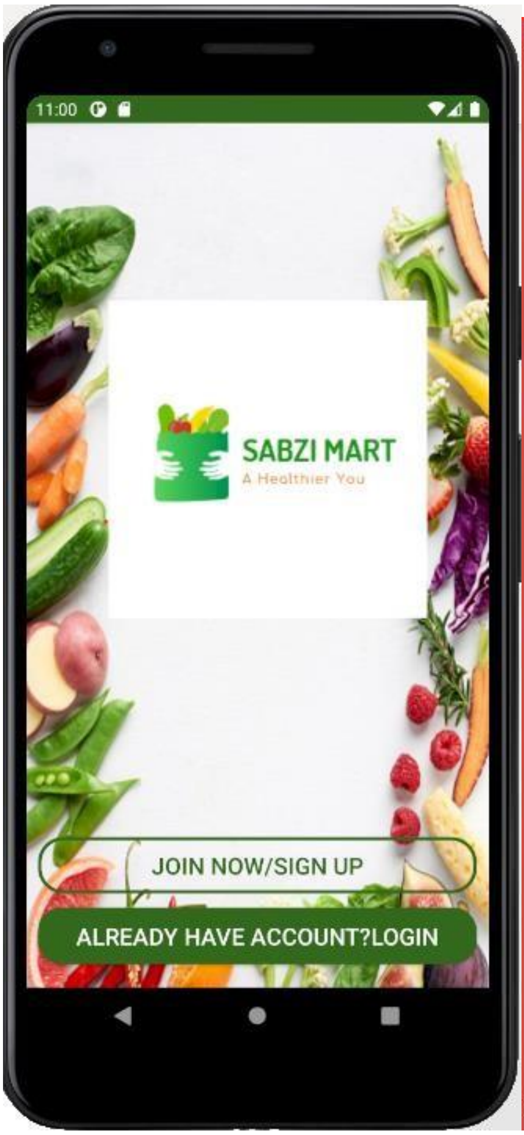
- Home activity is open where recycler view shows the product and on select certain product productdetail activity is open.
- Here user can select the quantity of product and add to cart. On add to cart product is added to cart
- User can go to cart and check out by providing detail about it info. The info would be saved in firebase under Order.

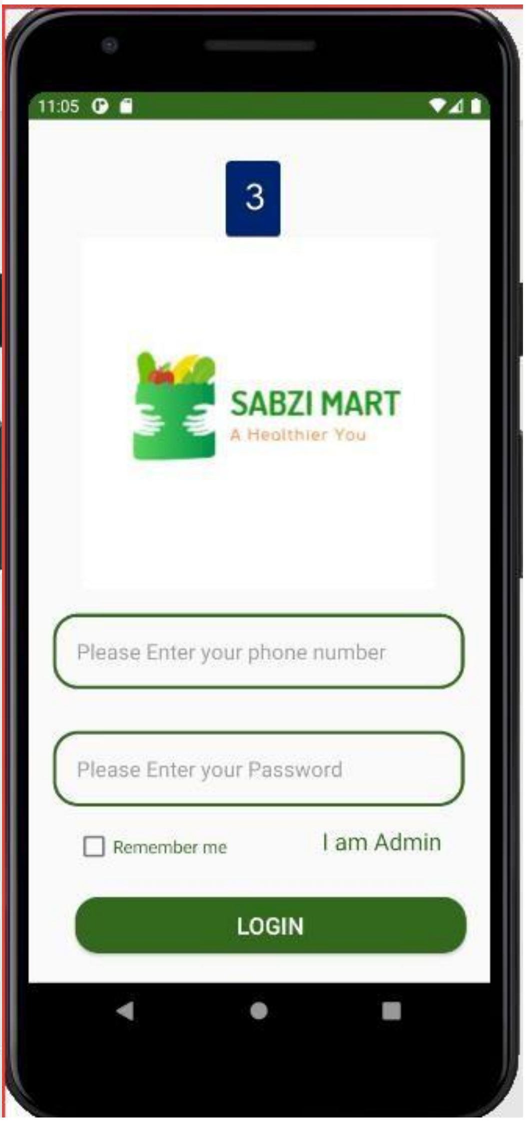
If login as admin

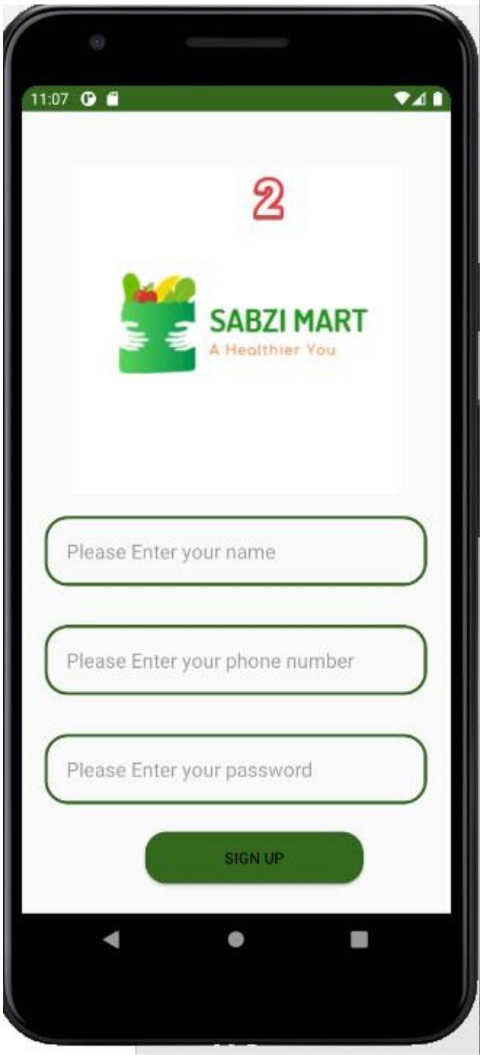
- Admins go the choice activity and choose to add a product or view order.
- On order saved in the database under Order child would be display using Recycler View

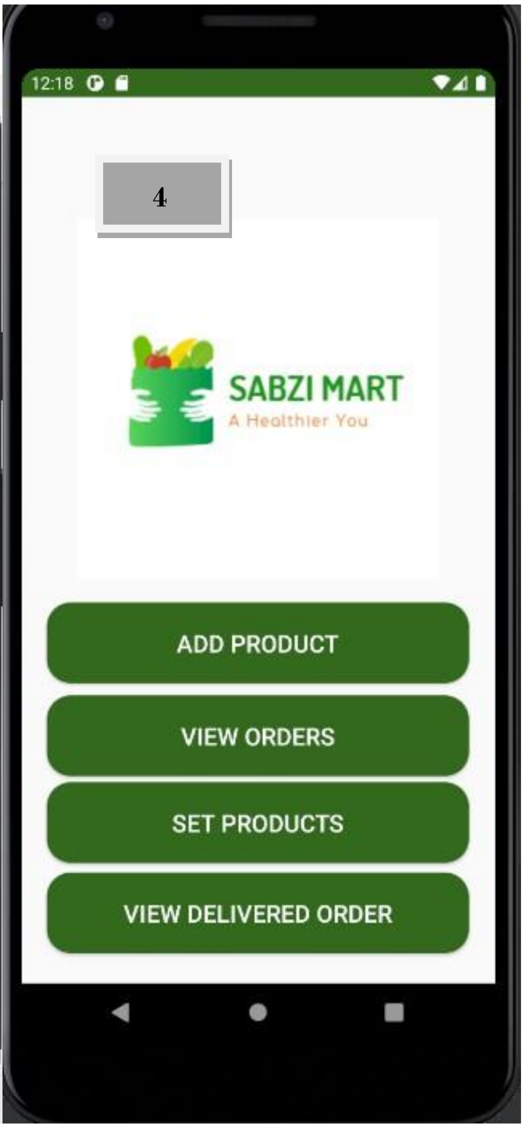
- On add a product admin select the product and put info to save in the database.

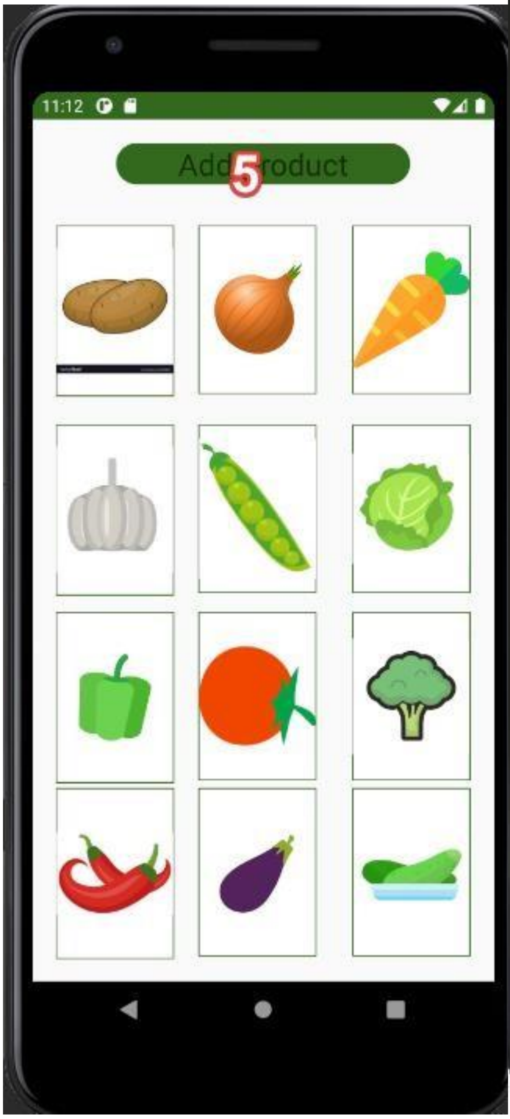
5. Results

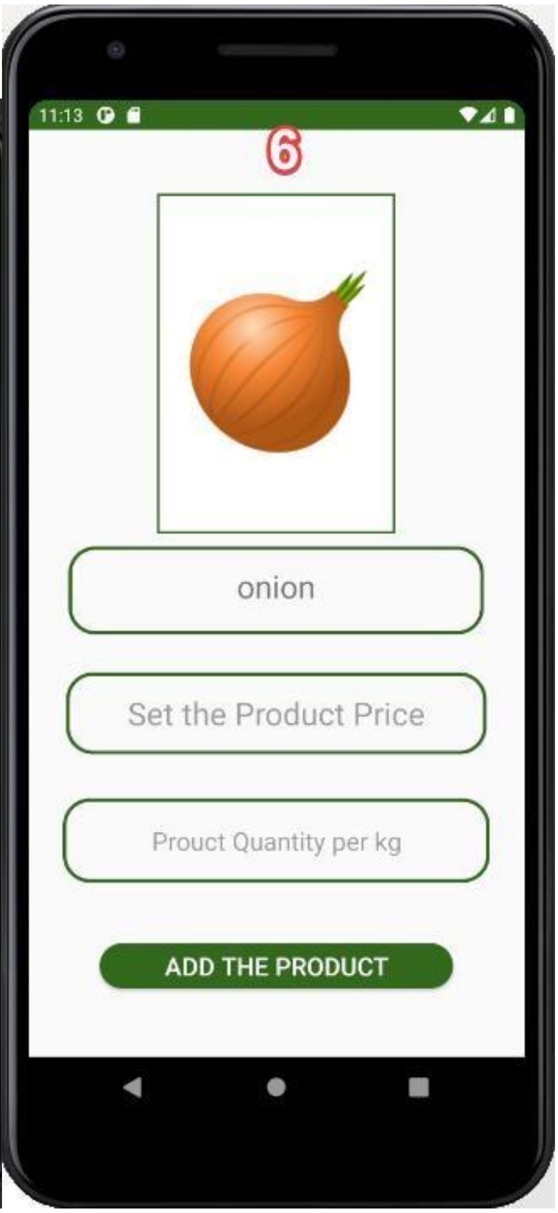





11:07
2
SABZI MART
A Healthier You
Please Enter your name
Please Enter your phone number
Please Enter your password
SIGN UP


12:18
4
SABZI MART
A Healthier You
ADD PRODUCT
VIEW ORDERS
SET PRODUCTS
VIEW DELIVERED ORDER

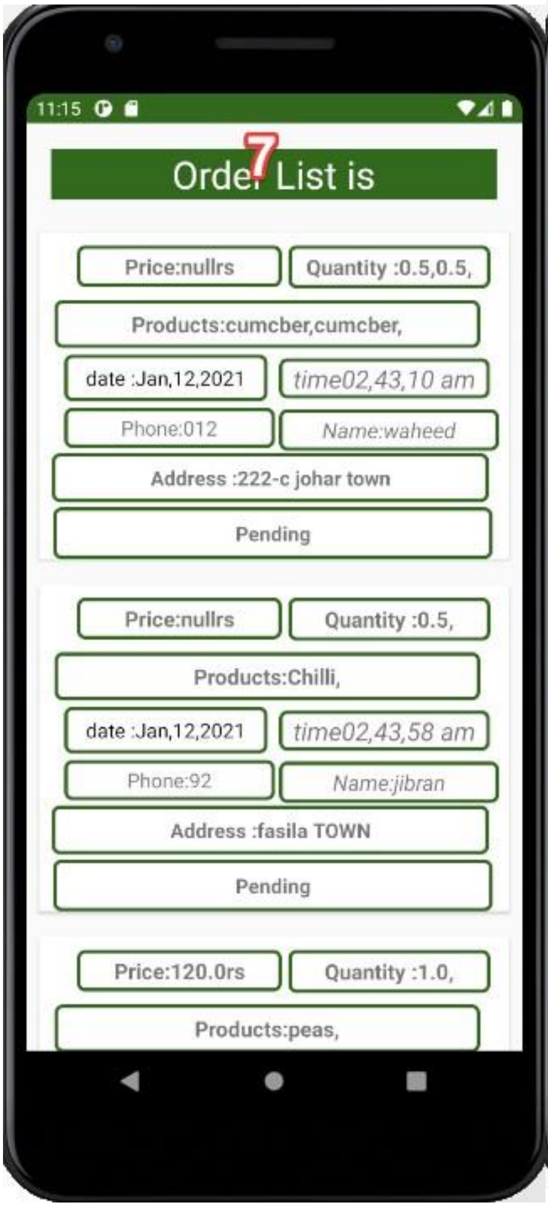




onion

Set the Product Price

Prouct Quantity per kg



11:15

Order List is

Price:nullrs

Quantity :0.5,0.5,

Products:cumcber,cumcber,

date :Jan,12,2021

time02,43,10 am

Phone:012

Name:waheed

Address :222-c johar town

Pending

Price:nullrs

Quantity :0.5,

Products:Chilli,

date :Jan,12,2021

time02,43,58 am

Phone:92

Name:jibran

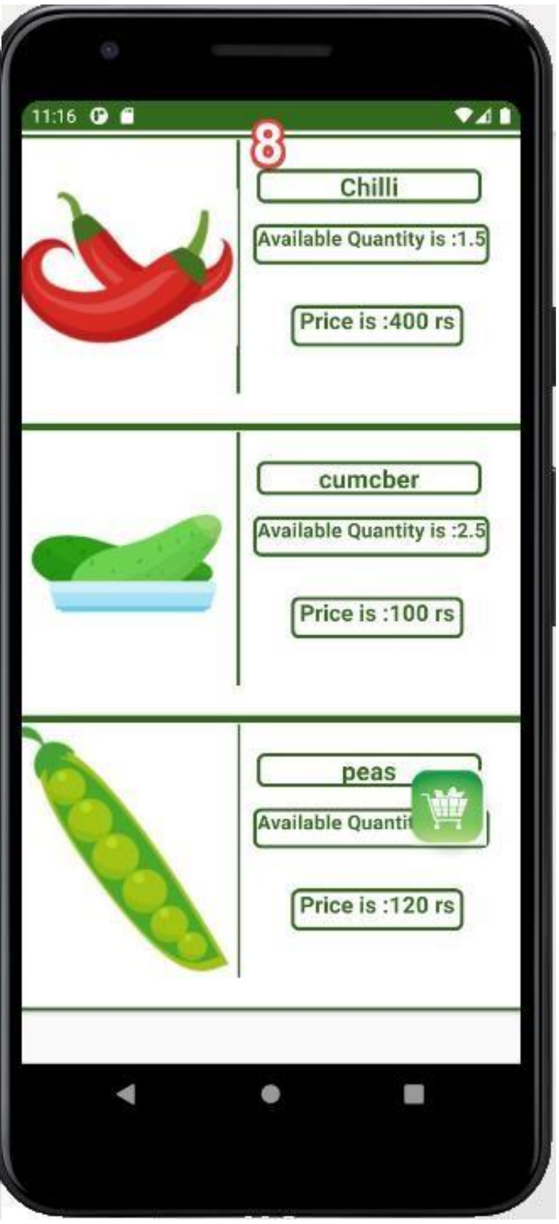
Address :fasila TOWN

Pending

Price:120.0rs

Quantity :1.0,

Products:peas,



11:16

Chilli

Available Quantity is :1.5

Price is :400 rs

cumcber

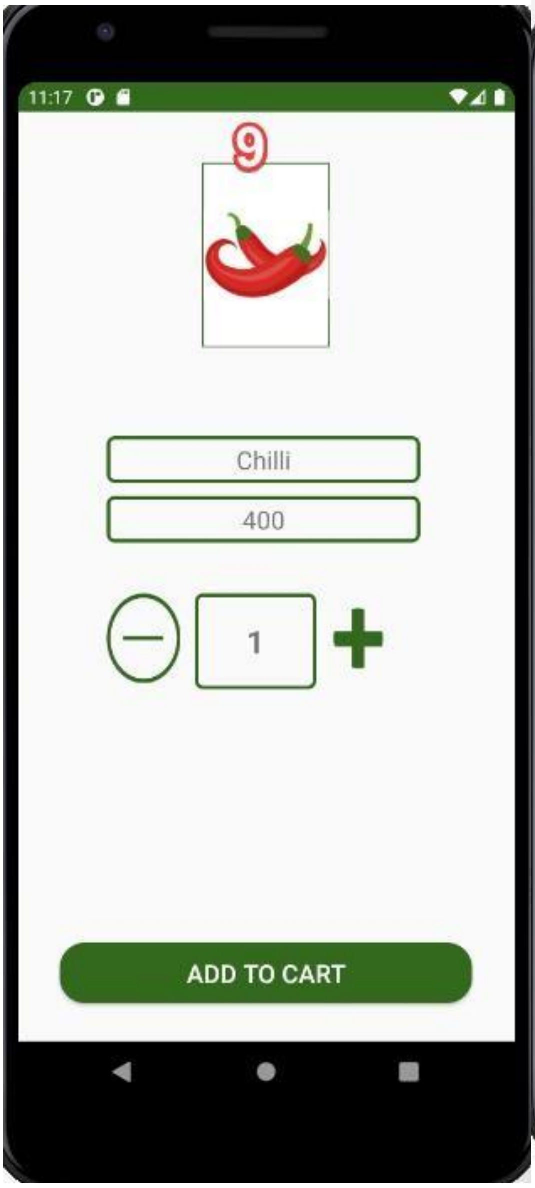
Available Quantity is :2.5

Price is :100 rs

peas

Available Quanti

Price is :120 rs

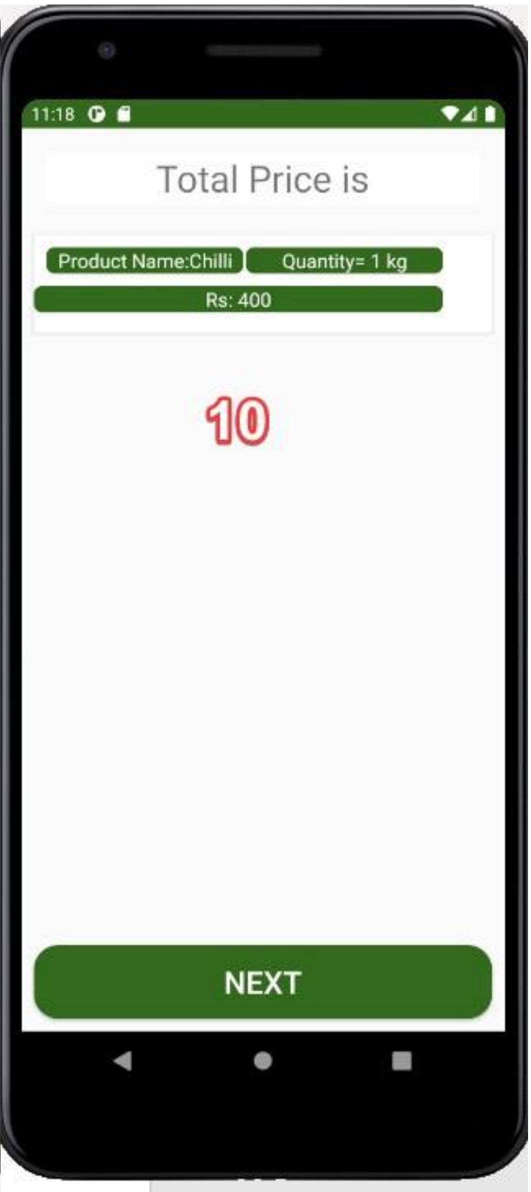


9

Chilli

400

1



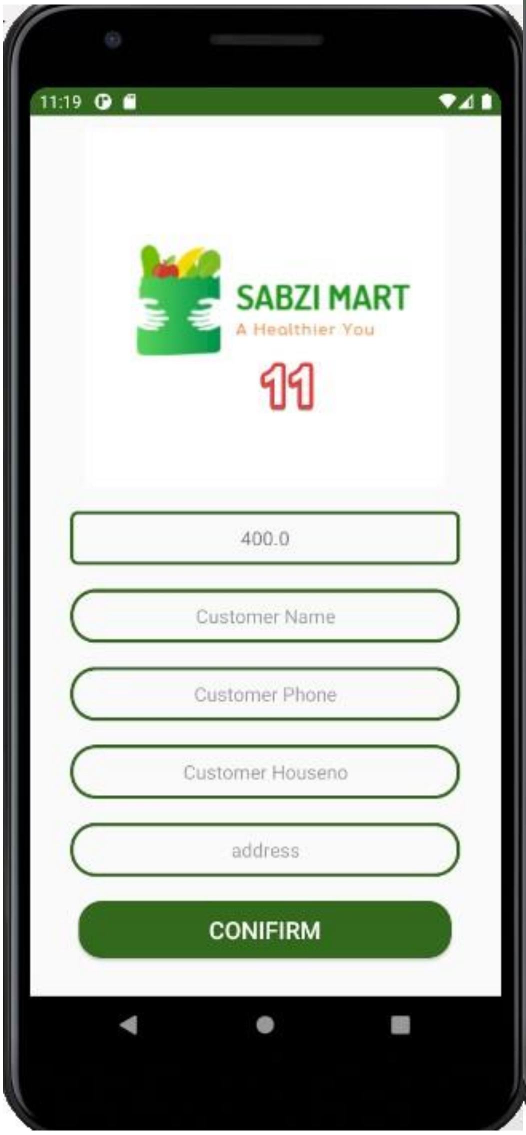
10

Total Price is

Product Name:Chilli

Quantity= 1 kg

Rs: 400



400.0

Customer Name

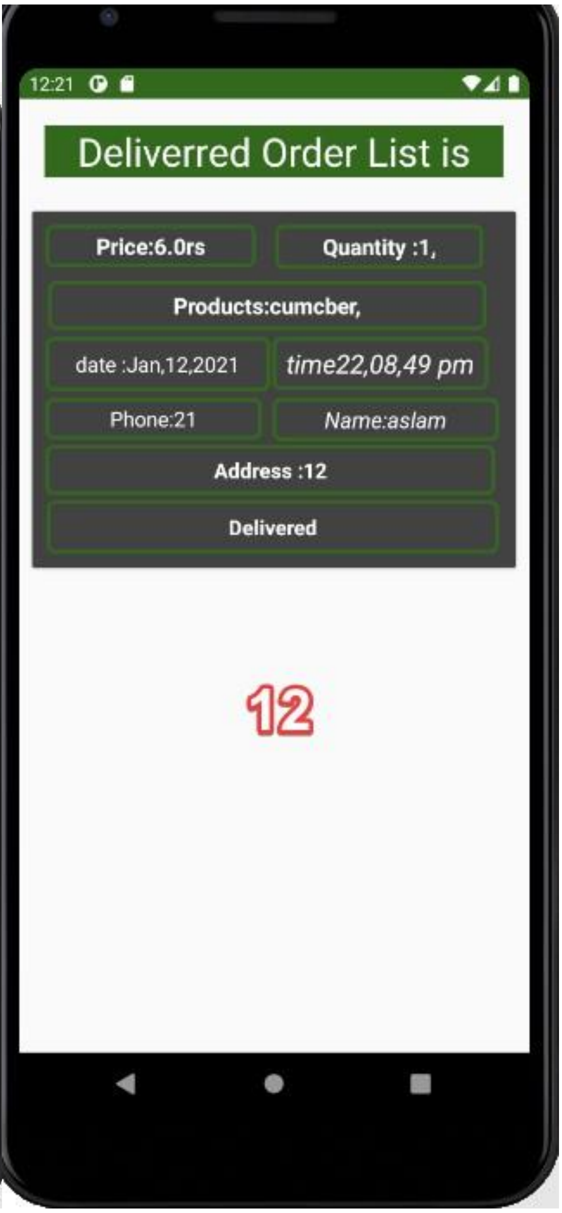
Customer Phone

Customer Houseno

address

CONIFIRM

11



Deliverred Order List is

Price:6.0rs Quantity :1,

Products:cumcber,

date :Jan,12,2021 time22,08,49 pm

Phone:21 Name:aslam

Address :12

Delivered

12

6. Code

MainActivity

```
public class MainActivity extends AppCompatActivity {

   @Override
   protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);        final
Button Signup=findViewById(R.id.joinnow);
Signup.setOnClickListener(new
View.OnClickListener() {            @Override
         public void onClick(View v) {
                Intent SignUp=new Intent(MainActivity.this,Signup.class);
                Toast.makeText(MainActivity.this,"New Activity
starting",Toast.LENGTH_SHORT).show();                 startActivity(SignUp);
            }
        });
        Button login=findViewById(R.id.ma_login);
        login.setOnClickListener(new View.OnClickListener() {
            @Override
public void onClick(View v) {

                Intent Logins=new Intent(MainActivity.this, Login.class);
startActivity(Logins);
            }
        });
    }
}
```

SignUp class

```
public class Signup extends AppCompatActivity {
    EditText user,password,phone;
    Button signup;
    ProgressDialog loading;
    String parentDb;
@Override
    protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_signup);

        user=(EditText)findViewById(R.id.name);
phone=findViewById(R.id.phone);
password=findViewById(R.id.password);
signup=findViewById(R.id.signup);         loading=new
ProgressDialog(this);
        //parentDb="Admins";
       signup.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

               CreateAccount();
            }
        });
    }
    private void CreateAccount(){
        String name=user.getText().toString();
        String pass=password.getText().toString();
String phone1=phone.getText().toString();
if(name.isEmpty())
        {
            Toast.makeText(this,"Please Enter the YOUR
name",Toast.LENGTH_SHORT).show();
        }
        else if(phone1.isEmpty())
        {
            Toast.makeText(this,"Please Enter your phone
number",Toast.LENGTH_SHORT).show();
        }
        else if(pass.isEmpty())
        {
            Toast.makeText(this,"Please enter your password",Toast.LENGTH_SHORT).show();
        }
else{
            loading.setTitle("Create Account");
            loading.setMessage("Please wait while we create your account");
loading.show();
            AddAccount(name,phone1,pass);
        }
    }
void AddAccount(final String name,final String Phone,final String Password){

        final DatabaseReference RootRef;
        RootRef= FirebaseDatabase.getInstance().getReference();
        RootRef.addListenerForSingleValueEvent(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
if(!(snapshot.child("Users").child(Phone).exists())){
HashMap<String,Object> userhash=new HashMap<>();
                    userhash.put("phone",Phone);
userhash.put("name",name);
userhash.put("password",Password);

RootRef.child("Users").child(Phone).updateChildren(userhash).addOnCompleteListener(new
OnCompleteListener<Void>() {
                        @Override
                        public void onComplete(@NonNull Task<Void> task) {
if(task.isSuccessful()){                                loading.dismiss();
Toast.makeText(Signup.this,"Congrats account is
created",Toast.LENGTH_SHORT).show();
                             Intent activity=new Intent(Signup.this,MainActivity.class);
startActivity(activity);
                         }
else{
```

```
                        loading.dismiss();
                        Toast.makeText(Signup.this,"Unable to Create Try
again",Toast.LENGTH_SHORT).show();
                      }
                    }
                  });
}                else{
                  Toast.makeText(Signup.this,"Account already exit on this phone
number",Toast.LENGTH_LONG).show();
                  Intent activity=new Intent(Signup.this,MainActivity.class);
startActivity(activity);
               }
           }

           @Override
           public void onCancelled(@NonNull DatabaseError error) {

           }
       });
}

}
```

Login

```
public class Login extends AppCompatActivity {
    EditText input_pw,input_phone;
    Button Logins;
    String Parentdb;
    TextView admin,notadmin;
    ProgressDialog loading;
    @Override
    protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_login);
init();
       //Admin Flag
        admin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
Parentdb="Admins";
admin.setVisibility(View.INVISIBLE);
notadmin.setVisibility(View.VISIBLE);
                Logins.setText("Login as Admin");
            }
        });
        //Not Admin text View
notadmin.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
Parentdb="Users";
notadmin.setVisibility(View.INVISIBLE);
admin.setVisibility(View.VISIBLE);
                Logins.setText("Login");
            }
        });
        //User Login Button
        Logins.setOnClickListener(new View.OnClickListener() {
            @Override              public void onClick(View v) {
loading.setTitle("Log into Account");
loading.setMessage("Please wait while we check crediablity of
account");                 loading.show();
                loading.setCancelable(false);
                Userlogin();
            }
        });
    }
    void Userlogin(){
        //password field text
        String pw=input_pw.getText().toString();
        //Phone Number Field Text
        String ph=input_phone.getText().toString();
if(pw.isEmpty()){
            Toast.makeText(Login.this,"Please Enter the
Password",Toast.LENGTH_SHORT).show();             loading.dismiss();
        }
        else if(ph.isEmpty()){
            Toast.makeText(Login.this,"Please Enter the Phone
Number",Toast.LENGTH_SHORT).show();             loading.dismiss();
       }
        //else check the user is user
exit         else{
           CheckUser(ph,pw);
        }

    }
    void CheckUser(final String phon,final
String Pw){         final DatabaseReference
RootRef;
        RootRef= FirebaseDatabase.getInstance().getReference();
        RootRef.addListenerForSingleValueEvent(new ValueEventListener() {
            @Override             public void
onDataChange(@NonNull DataSnapshot snapshot) {
                //User exit or not                 if
(snapshot.child(Parentdb).child(phon).exists()) {
                    //Data into user Model Class
                    Users data=snapshot.child(Parentdb).child(phon).getValue(Users.class);
if(data.getPhone().equals(phon)){
                        {
                            if(data.getPassword().equals(Pw)){
                               Toast.makeText(Login.this,"Logined
Successfully",Toast.LENGTH_SHORT).show();                              loading.dismiss();
                            if(Parentdb.equals("Admins")){
                                Intent addProduct=new Intent(Login.this,admin_opion.class);
startActivity(addProduct);
                            }
                            else { Intent home=new Intent(Login.this,HomeActivity.class);
home.putExtra("phone",phon);startActivity(home);}
                            }
                            //Check Password
                            else{ Toast.makeText(Login.this,"Password
Incorrect",Toast.LENGTH_SHORT).show();
                            loading.dismiss();}
                        } } }
else{
                    Toast.makeText(Login.this,"No Such UsserExit
",Toast.LENGTH_SHORT).show();                     loading.dismiss();
                }

        }
            @Override             public void
onCancelled(@NonNull DatabaseError error) {

            }
       });
    }
void init(){
        //Password
input_pw=findViewById(R.id.pw);
        //Phone Number
input_phone=(findViewById(R.id.Phone_no));
        //Login Button
        Logins=findViewById(R.id.login);
loading=new ProgressDialog(this);
Parentdb="Users";
admin=findViewById(R.id.adminflag);
```

```
        //not admin text view
notadmin=findViewById(R.id.notAdmin);

    }
Admin_Option public class admin_opion
extends AppCompatActivity {

    @Override     protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_admin_opion);
Button
Add_Product=findViewById(R.id.bt_Add_Product);
        Button view_order=findViewById(R.id.newOrder_bt);
        Button view_product=findViewById(R.id.set_product_bt);
        Button delivery_order=findViewById(R.id.delivered_button);
        Add_Product.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent addproduct=new Intent(admin_opion.this,Add_Product.class);
startActivity(addproduct);
            }
        });
        view_order.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent viewor=new Intent(admin_opion.this,orders.class);
startActivity(viewor);
            }
        });
        view_product.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent viewor=new Intent(admin_opion.this,HomeActivity.class);
viewor.putExtra("category","admin");                  startActivity(viewor);

            }
        });
        delivery_order.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent viewor=new Intent(admin_opion.this,deliveredorder.class);
startActivity(viewor);
            }
        });
    }
}
```

Add Product

```
public class Add_Product extends AppCompatActivity {
    ImageView imageView[]=new ImageView[12];
    @Override     protected void
onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_add__produ
ct);        init();

    }
void init(){

        imageView[0]=(ImageView)findViewById(R.id.potato);
imageView[1]=(ImageView)findViewById(R.id.onion);
imageView[2]=(ImageView)findViewById(R.id.carrot);
imageView[3]=(ImageView)findViewById(R.id.garlic);
imageView[4]=(ImageView)findViewById(R.id.peas);
imageView[5]=(ImageView)findViewById(R.id.Brussel);
imageView[6]=(ImageView)findViewById(R.id.green);
imageView[7]=(ImageView)findViewById(R.id.tomato);
imageView[8]=(ImageView)findViewById(R.id.Gobli);
imageView[9]=(ImageView)findViewById(R.id.chilli);
imageView[10]=(ImageView)findViewById(R.id.eggplant);
imageView[11]=(ImageView)findViewById(R.id.cumber);
    }
    public void ProductSelect(View v){


v.setDrawingCacheEnabled(true);
Bitmap b=v.getDrawingCache();
Intent Product_Detail=new
Intent(Add_Product.this,Product_Set.
class);
        //send the Vegatable name
        Product_Detail.putExtra("name",v.getTag().toString());
        //Vegtable Image
        Product_Detail.putExtra("image",b);
startActivity(Product_Detail);
    }
}
```

Product_Set

```
public class Product_Set extends AppCompatActivity {
    EditText Product_price,Product_quantity;
    TextView Product_name;
    String P_NAME,P_PRICE,P_QUANTITY,ImageUrl;
    Button Add_To_item,Check_order;
    ImageView mainImage;
    Uri imageUri;
    Boolean flag=false;
     StorageReference ProductImagesRef;
     ProgressDialog loading;
     DatabaseReference ProductReference;
    @Override     protected void
onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_product__set
);         init();
        //get the data from previous activity
        Intent data=getIntent();
        Product_name.setText(data.getStringExtra("name"));
        //get images
       Bitmap bitmap=data.getParcelableExtra("image");
       //set image
mainImage.setImageBitmap(bitmap);
mainImage.setClickable(true);
mainImage.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                ChoseImage(); }});
        Add_To_item.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                StoreInformation();
            }
        });

    }
    void StoreInformation() {
        P_NAME = Product_name.getText().toString();
        P_PRICE =
Product_price.getText().toString();
P_QUANTITY =
Product_quantity.getText().toString();         if
(P_PRICE.isEmpty()) {
            Toast.makeText(Product_Set.this, "Please fill Price field",
Toast.LENGTH_SHORT).show();
        } else if (P_QUANTITY.isEmpty()) {  Toast.makeText(Product_Set.this, "Please
fill Quantity", Toast.LENGTH_SHORT).show();}
        else{
            Uploadimage();
        }
    }
    void Uploadimage(){         loading.setTitle("Image
Uploading");         loading.setMessage("please Wait while
we upload Product image");         loading.show();
        //Create name for the branch         final StorageReference
filePath=ProductImagesRef.child(P_NAME+".png");
        //Put the data in the form
bytes         final UploadTask
uploadTask;



            mainImage.setDrawingCacheEnabled(true);
mainImage.buildDrawingCache();
            Bitmap bitmap = ((BitmapDrawable)
mainImage.getDrawable()).getBitmap();
ByteArrayOutputStream baos = new ByteArrayOutputStream();
bitmap.compress(Bitmap.CompressFormat.JPEG, 100, baos);
byte[] data = baos.toByteArray();
        filePath.putBytes(data).continueWithTask(new
Continuation<UploadTask.TaskSnapshot, Task<Uri>>() {
            @Override             public Task<Uri> then(@NonNull
Task<UploadTask.TaskSnapshot> task) throws Exception {                 if
(!task.isSuccessful()) {                     throw task.getException();
                }
                return filePath.getDownloadUrl();
            }
        }).addOnCompleteListener(new OnCompleteListener<Uri>() {
            @Override             public void
onComplete(@NonNull Task<Uri> task) {
if (task.isSuccessful()) {
                    ImageUrl = task.getResult().toString();
                    Toast.makeText(Product_Set.this, "Image Uploading",
Toast.LENGTH_SHORT).show();                     savedata();
                } else {
                    Toast.makeText(Product_Set.this, "upload failed: " +
task.getException().getMessage(), Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
    void savedata(){
        HashMap<String,Object> product=new HashMap<>();
product.put("product_name",P_NAME);
product.put("image",ImageUrl);
product.put("product_price",P_PRICE);
product.put("product_quantity",P_QUANTITY);

ProductReference.child(P_NAME).updateChildren(product).addOnCompleteListener
(new
OnCompleteListener<Void>() {
            @Override             public void
onComplete(@NonNull Task<Void> task) {
if(task.isSuccessful()){
                    loading.dismiss();Toast.makeText(Product_Set.this, "data Uploaded",
Toast.LENGTH_SHORT).show();}
                else{
                    loading.dismiss();Toast.makeText(Product_Set.this,
task.getException().toString(), Toast.LENGTH_SHORT).show();}
            }
        });
    }
    void ChoseImage(){
        Intent gallery = new Intent();
gallery.setAction(Intent.ACTION_GET_CONTENT
);         gallery.setType("image/*");
startActivityForResult(gallery, 1);
    }

    @Override     protected void onActivityResult(int requestCode, int
resultCode, @Nullable Intent data) {
super.onActivityResult(requestCode, resultCode, data);
        if (requestCode==1  &&  resultCode==RESULT_OK  &&  data!=null)
        {            flag=true;
imageUri = data.getData();
mainImage.setImageURI(imageUri)
;
        }
    }

    void init(){
        Product_name=findViewById(R.id.edn_name);
        Product_quantity=findViewById(R.id.ed_quanlity);
        Product_price=findViewById(R.id.ed_price);
Add_To_item=findViewById(R.id.bt_add);
mainImage=findViewById(R.id.product_image);
loading=new ProgressDialog(this);
        ProductImagesRef =
FirebaseStorage.getInstance().getReference().child("Product Images");
        ProductReference =
FirebaseDatabase.getInstance().getReference().child("Products");
```

```
    }
}
```

Orders

```
public class orders extends AppCompatActivity {
    RecyclerView recyclerView;
    RecyclerView.LayoutManager layoutManager;
    String PHONE;     @Override     protected
void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_orders);
init();
    }
    @Override
protected void
onStart() {
super.onStart();
        final DatabaseReference orderListReference=
FirebaseDatabase.getInstance().getReference().child("Order");
        FirebaseRecyclerOptions<Order>option=new
FirebaseRecyclerOptions.Builder<Order>().setQuery(orderListReference,Order.class
).build();
        FirebaseRecyclerAdapter<Order, OrderViewHolder> adapter= new
FirebaseRecyclerAdapter<Order, OrderViewHolder>(option) {
            @NonNull
@Override
            public OrderViewHolder onCreateViewHolder(@NonNull ViewGroup parent,
int viewType) {                View view =
LayoutInflater.from(parent.getContext()).inflate(R.layout.admin_order, parent, false);
                OrderViewHolder holder = new
OrderViewHolder(view);                return holder;
            }

            @Override             protected void onBindViewHolder(@NonNull
OrderViewHolder orderViewHolder, int i,
@NonNull final Order order) {
orderViewHolder.admin_address.setText("Address :"+order.getAddress());
orderViewHolder.admin_pquantity.setText("Quantity
:"+order.getProduct_Quantity());
orderViewHolder.admin_date.setText("date :"+order.getDate());
orderViewHolder.admin_pname.setText("Products:"+order.getProduct_NAME
S());                orderViewHolder.admin_time.setText("time"+order.getTime());
orderViewHolder.admin_price.setText("Price:"+order.getPrice()+"rs");
orderViewHolder.admin_phone.setText("Phone:"+order.getPhone());
orderViewHolder.admin_Name.setText("Name:"+order.getName());
orderViewHolder.state.setText(order.getState());
PHONE=getRef(i).getKey();
orderViewHolder.itemView.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        AlertDialog.Builder builder=new
AlertDialog.Builder(orders.this);                        CharSequence
options[]=new CharSequence[]{"Delivered","pending"};
builder.setTitle("Order ITEMS");
builder.setItems(options, new DialogInterface.OnClickListener() {
                            @Override                          public void
onClick(DialogInterface dialog, int which) {
if(which==1) {

                                }
else if(which==0){

orderListReference.child("Order").child(PHONE).removeValue().addOnCompleteList
ener(new
OnCompleteListener<Void>() {
                                        @Override
public void onComplete(@NonNull Task<Void> task) {
if(task.isSuccessful()){

                                                Toast.makeText(orders.this,"Order remove from the
Pending Order List successfully"+PHONE,Toast.LENGTH_SHORT).show();
                                                finish();
startActivity(getIntent());
                                            } }}); } }});
builder.show(); }

                });



            }
        };
        recyclerView.setAdapter(adapter);
adapter.startListening();

    }
void init(){
        recyclerView=findViewById(R.id.order_recyclerview);
layoutManager=new LinearLayoutManager(this);
recyclerView.setLayoutManager(layoutManager);
    }
}
```

Product_detail public class productdetail

```
extends AppCompatActivity {
   TextView product_dname,product_dprice,getProduct_dquantity;;
   ImageView image_d;
   Double Max;
   String PHONE;
   String Product_Name;
   Button plus,negative,add_to_cart;
    @Override     protected void
onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_productdetail
);         init();
        Product_Name=getIntent().getStringExtra("product");
PHONE=getIntent().getStringExtra("phone");
get_Product_Detail(Product_Name);
plus.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Double
a=Double.parseDouble(getProduct_dquantity.getText().toString());
a=a+0.25;                  if(a<=Max)
getProduct_dquantity.setText(String.valueOf(a));                  else
                    Toast.makeText(productdetail.this,"Value cannot get here than
avaiable",Toast.LENGTH_SHORT).show();

            }
        });

        negative.setOnClickListener(new View.OnClickListener() {
            @Override
public void onClick(View v) {
Double
a=Double.parseDouble(getPr
oduct_dquantity.getText().to
String());                  a=a-0.25;
if(a<0){
                    Toast.makeText(productdetail.this,"Value cannot get lower than
zero",Toast.LENGTH_SHORT).show();
                }
else
                getProduct_dquantity.setText(String.valueOf(a));
            }
        });
        //addtocart
add_to_cart.setOnClickListener(new
View.OnClickListener() {
```

```
            @Override
public void onClick(View v) {
add_to_cart_list();
            }
        });
    }
    void add_to_cart_list(){
        String current_date,current_Time;
        Calendar calfordate=Calendar.getInstance();
        SimpleDateFormat currentdate= new SimpleDateFormat("MMM,dd,yyy");
current_date=currentdate.format(calfordate.getTime());

        SimpleDateFormat currenttime= new SimpleDateFormat("HH,mm,ss a");
current_Time=currenttime.format(calfordate.getTime());
       final  DatabaseReference
cartlistRef=FirebaseDatabase.getInstance().getReference().child("Cart
List");
        final HashMap<String,Object> cartdMap=new
HashMap<>();         cartdMap.put("date",current_date);
cartdMap.put("time",current_Time);
cartdMap.put("P_name",product_dname.getText().toString());
cartdMap.put("P_price",product_dprice.getText().toString());
cartdMap.put("P_quantity",getProduct_dquantity.getText().toS
tring());         cartlistRef.child("User
View").child(PHONE).child(product_dname.getText().toString()).updateChildren(cartd
Map)
                .addOnCompleteListener(new OnCompleteListener<Void>() {
                    @Override                     public void
onComplete(@NonNull Task<Void> task) {
                        //for admin purpose
if(task.isSuccessful()){
cartlistRef.child("Admin
View").child(PHONE).child(product_dname.getText().toString()).updateChildren(cartd
Map)
                                    .addOnCompleteListener(new OnCompleteListener<Void>() {
                                        @Override
public void onComplete(@NonNull Task<Void> task) {
if(task.isSuccessful()) {
                                                Toast.makeText(productdetail.this, "Product add to
CART LISt",
Toast.LENGTH_SHORT).show();
                                                Intent bact_home=new
Intent(productdetail.this,HomeActivity.class);
bact_home.putExtra("phone",PHONE);
startActivity(bact_home);
                                            }
                                        }
                                    });
                        }
                    }
                });

    }
    void get_Product_Detail(final String Key){
DatabaseReference productReference=
FirebaseDatabase.getInstance().getReference().child("Products");
productReference.child(Key).addValueEventListener(new ValueEventListener() {
            @Override             public void
onDataChange(@NonNull DataSnapshot snapshot) {
if(snapshot.exists()) {
                    Products product_d = snapshot.getValue(Products.class);
product_dname.setText(Key);
                    product_dprice.setText(product_d.getProduct_price());
                    Max = Double.parseDouble(product_d.getProduct_quantity());
                    Picasso.get().load(product_d.getImage()).into(image_d);
                }
            }

            @Override             public void
onCancelled(@NonNull DatabaseError error) {

            }
        });



    }
void init(){
        product_dname=findViewById(R.id.product_detail_name);
product_dprice=findViewById(R.id.product_detail_price);
image_d=findViewById(R.id.prouduct_detail_image);
getProduct_dquantity=findViewById(R.id.avaiable);
plus=findViewById(R.id.plus_button);
negative=findViewById(R.id.negative_button);
add_to_cart=findViewById(R.id.add_to_cart);
    }
}
```

Cart

```
public class cart extends AppCompatActivity {
    TextView cart_total_price;
    String PHONE;
    Button cart_next_btn;
RecyclerView
```

```
cart_recycler_view;     double
Total_price=0;
ArrayList<String> ProductList;
   ArrayList<String> quantityList;
   RecyclerView.LayoutManager layoutManager;
   @Override     protected void
onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_cart);
      init();
      PHONE=getIntent().getStringExtra("phone");
      cart_next_btn.setOnClickListener(new View.OnClickListener() {
         @Override
public void onClick(View v) {
          cart_total_price.setText("Total Price
="+Total_price);                Intent conifirm=new
Intent(cart.this,conifirmorder.class);
conifirm.putExtra("total",String.valueOf(Total_price));
conifirm.putExtra("phone",PHONE);
conifirm.putExtra("Product List",ProductList);
conifirm.putExtra("quantity list",quantityList);
startActivity(conifirm);

         }
      });
   }
   @Override
   protected void onStart() {

      super.onStart();
      final DatabaseReference cartListReference=
FirebaseDatabase.getInstance().getReference().child("Cart List");
     FirebaseRecyclerOptions<Cart> options=new
     FirebaseRecyclerOptions.Builder<Cart>().setQuery(cartListReference.child("Use
     r
View").child(PHONE),Cart.class).build();
      FirebaseRecyclerAdapter<Cart, CartViewHolder> adapter= new
FirebaseRecyclerAdapter<Cart,
CartViewHolder>(options) {
         @NonNull
@Override
         public CartViewHolder onCreateViewHolder(@NonNull ViewGroup parent,
int viewType) {                View view =
LayoutInflater.from(parent.getContext()).inflate(R.layout.cart_items, parent, false);
             CartViewHolder holder = new
CartViewHolder(view);                return holder;
```

```
            }
            @Override             protected void onBindViewHolder(@NonNull final
CartViewHolder cartViewHolder, int i,
@NonNull final Cart cart) {
cartViewHolder.txtProduct_name.setText("Product
Name:"+cart.getP_name());
cartViewHolder.txtProduct_price.setText("Rs: "+cart.getP_price());
                cartViewHolder.txtProduct_quantity.setText("Quantity=
"+cart.getP_quantity()+" kg");                 ProductList.add(cart.getP_name());
quantityList.add(cart.getP_quantity());
cart_next_btn.setClickable(true);
                double
singlevalue=Double.valueOf(cart.getP_price())*Double.valueOf(cart.getP_quantity()
);                 Total_price=Total_price+singlevalue;
cartViewHolder.itemView.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        AlertDialog.Builder builder=new
AlertDialog.Builder(cart.this);                        CharSequence
options[]=new CharSequence[]{"Edit","discart"};
builder.setTitle("CART ITEMS");
builder.setItems(options, new DialogInterface.OnClickListener() {
                            @Override                             public void
onClick(DialogInterface dialog, int which) {
if(which==0) {
                                    Intent edit_activity=new
Intent(cart.this,productdetail.class);
edit_activity.putExtra("product",cart.getP_name());
edit_activity.putExtra("phone",PHONE);
startActivity(edit_activity);
                                }
else if(which==1){
                                    cartListReference.child("User
View").child(PHONE).child(cart.getP_name()).removeValue().addOnCompleteListen
er(new
OnCompleteListener<Void>() {
                                        @Override
public void onComplete(@NonNull Task<Void> task) {
if(task.isSuccessful()){
                                                Toast.makeText(cart.this,"item remove from the cart
successfully",Toast.LENGTH_SHORT).show();
                                                Intent home_activity=new
Intent(cart.this,HomeActivity.class);
home_activity.putExtra("phone",PHONE);
startActivity(home_activity);
                                            } }}); } }});
builder.show(); }
```

```
            });

         }
      };
      cart_recycler_view.setAdapter(adapter);
adapter.startListening();
   }
    void init(){
cart_total_price=findViewById(R.id.cart_total_price
);
cart_next_btn=findViewById(R.id.cart_next_bt);
cart_recycler_view=findViewById(R.id.cart_recycle
rview);        layoutManager=new
LinearLayoutManager(this);
cart_recycler_view.setLayoutManager(layoutMana
ger);        ProductList=new ArrayList<>();
quantityList=new ArrayList<>();



   }
}
```

ConifrimOrder

```
public class conifirmorder extends AppCompatActivity {
    TextView conf_total_price;
    EditText conf_cname,conf_cphone,conf_caddress,conf_houseno;
   Button Conifirm_bt;
   String PHONE,cust_Name,cust_phone,cust_address,cust_houseno;
   ArrayList<String> P_list,P_quantity;
   Boolean flag=false;     @Override
protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_conifirmorde
r);        init();
     get_intent_data();
      Conifirm_bt.setOnClickListener(new View.OnClickListener() {
         @Override
public void onClick(View v) {
validatedata();
         }
      });

   }
   void validatedata() {
cust_Name=conf_cname.getText().toString();
cust_address=conf_caddress.getText().toString();
cust_phone=conf_cphone.getText().toString();
cust_houseno=conf_houseno.getText().toString();
if(cust_Name.isEmpty()){
         Toast.makeText(conifirmorder.this,"PLEASE FILL ALL THE
FIELDS",Toast.LENGTH_SHORT).show();
      }
      else if(cust_address.isEmpty()){
         Toast.makeText(conifirmorder.this,"PLEASE FILL ALL THE
FIELDS",Toast.LENGTH_SHORT).show();
      }
      else if(cust_phone.isEmpty()){
         Toast.makeText(conifirmorder.this,"PLEASE FILL ALL THE
FIELDS",Toast.LENGTH_SHORT).show();
      }
      else if(cust_houseno.isEmpty()){
         Toast.makeText(conifirmorder.this,"PLEASE FILL ALL THE
FIELDS",Toast.LENGTH_SHORT).show();
      }
else{
         ConifirmOrder(cust_Name,cust_address,cust_phone,cust_houseno);
      }
   }
   void get_intent_data(){
      P_list = (ArrayList<String>) getIntent().getSerializableExtra("Product
List");         P_quantity = (ArrayList<String>)
getIntent().getSerializableExtra("quantity list");
conf_total_price.setText(getIntent().getStringExtra("total"));
      PHONE=getIntent().getStringExtra("phone");
   }

   private void ConifirmOrder(final String customer_name, final String
customer_address, final String customer_phone,final String customer_houseno) {

     String current_date,current_Time;
      Calendar calfordate=Calendar.getInstance();
      SimpleDateFormat currentdate= new SimpleDateFormat("MMM,dd,yyy");
current_date=currentdate.format(calfordate.getTime());

      SimpleDateFormat currenttime= new
SimpleDateFormat("HH,mm,ss a");
current_Time=currenttime.format(calfordate.getTime());
final DatabaseReference orderlistRef=
FirebaseDatabase.getInstance().getReference().child("Order").child(cu
stomer_phone);         final HashMap<String,Object> orderdMap=new
HashMap<>();         orderdMap.put("Date",current_date);
orderdMap.put("Time",current_Time);
orderdMap.put("Product_NAMES",get_List_String(P_list));
orderdMap.put("Product_Quantity",get_List_String(P_quantity));
orderdMap.put("Address",customer_address);
orderdMap.put("Phone",customer_phone);
orderdMap.put("House",customer_houseno);
orderdMap.put("Price",conf_total_price.getText().toString());
orderdMap.put("Name",customer_name);
orderdMap.put("State","Pending");
      orderlistRef.updateChildren(orderdMap).addOnCompleteListener(new
OnCompleteListener<Void>() {
         @Override            public void
onComplete(@NonNull Task<Void> task) {
if(task.isSuccessful())
            {
               FirebaseDatabase.getInstance().getReference("Cart List").child("User
View").removeValue().
                     addOnCompleteListener(new OnCompleteListener() {
                        @Override
public void onComplete(@NonNull Task task) {
if(task.isSuccessful()) {
                              Toast.makeText(conifirmorder.this, "Order Placed ",
LENGTH_LONG).show();                              int i=0;
for (String a:P_list) {

updatedb(a,i);
i++;
                              } }

                           }
                     });

}          }
     });

   }
   void updatedb(final String a, final int i)
   {
      final DatabaseReference updateproduct =
FirebaseDatabase.getInstance().getReference("Products").
child(a);         final  Double quantity_sub =
Double.valueOf(P_quantity.get(i));
      updateproduct.addListenerForSingleValueEvent(new ValueEventListener() {
          @Override             public void
onDataChange(@NonNull DataSnapshot snapshot) {

              if(!flag){
              Double quantity_total =
Double.valueOf(snapshot.child("product_quantity").getValue().toSt
ring());                Double quantity_real = quantity_total -
quantity_sub;

updateproduct.child("product_quantity").setValue(quantity_real.toString()).addOnCo
mpleteListener(new OnCompleteListener<Void>() {
                 @Override                    public void
onComplete(@NonNull Task<Void> task) {
if (task.isSuccessful()) {
if(i==(P_list.size()-1)) {
                            Toast.makeText(conifirmorder.this, "Product Data Updated ",
LENGTH_LONG).show();
                            Intent home = new Intent(conifirmorder.this, HomeActivity.class);
home.putExtra("phone", PHONE);
                            home.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
                            startActivity(home);
                        }
flag = true;

                    } else {
                        Toast.makeText(conifirmorder.this, task.getException().toString(),
LENGTH_LONG).show();
                    }
                }
              });
              }
              }



          @Override             public void
onCancelled(@NonNull DatabaseError error) {
Toast.makeText(conifirmorder.this, error.toString(),
LENGTH_LONG).show();
          }
      });
}

   String
get_List_String(ArrayList<String>list){
String List_String="";        int i=0;
```

```
        while(i!=list.size()) {
            List_String = List_String +
list.get(0).toString()+",";             i++;
        }
        return List_String;
    }     void init(){
conf_total_price=findViewById(R.id.conifirm_total_
price);
conf_cname=findViewById(R.id.conifirm_Cname);
conf_caddress=findViewById(R.id.conifirm_Caddr
ess);
conf_cphone=findViewById(R.id.conifirm_Cphone)
;
conf_houseno=findViewById(R.id.conifirm_Chous
eno);
        Conifirm_bt=findViewById(R.id.bt_confirm);
        P_list=new ArrayList<>();
        P_quantity=new ArrayList<>();
    }
}
```

deliveredorder

```
public class deliveredorder extends AppCompatActivity {
    RecyclerView recyclerView_del;
    RecyclerView.LayoutManager layoutManager;
    @Override     protected void
onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_deliveredord
er);         init();
    }
    @Override
protected void
onStart() {
super.onStart();
        final DatabaseReference orderListReference=
FirebaseDatabase.getInstance().getReference().child("DELIVERED Order");
        FirebaseRecyclerOptions<Order> option=new
FirebaseRecyclerOptions.Builder<Order>().setQuery(orderListReference,Order.class
).build();
        FirebaseRecyclerAdapter<Order, OrderViewHolder> adapter= new
FirebaseRecyclerAdapter<Order, OrderViewHolder>(option) {
            @NonNull
@Override
            public OrderViewHolder onCreateViewHolder(@NonNull ViewGroup parent,
int viewType) {                 View view =
LayoutInflater.from(parent.getContext()).inflate(R.layout.admin_order, parent, false);
```

```
                OrderViewHolder holder = new
OrderViewHolder(view);                 return holder;
            }

            @Override             protected void onBindViewHolder(@NonNull
OrderViewHolder orderViewHolder, int i,
@NonNull final Order order) {
orderViewHolder.admin_address.setText("Address :"+order.getAddress());
orderViewHolder.admin_pquantity.setText("Quantity
:"+order.getProduct_Quantity());
orderViewHolder.admin_date.setText("date :"+order.getDate());
orderViewHolder.admin_pname.setText("Products:"+order.getProduct_NAME
S());                 orderViewHolder.admin_time.setText("time"+order.getTime());
orderViewHolder.admin_price.setText("Price:"+order.getPrice()+"rs");
orderViewHolder.admin_phone.setText("Phone:"+order.getPhone());
orderViewHolder.admin_Name.setText("Name:"+order.getName());
orderViewHolder.state.setText(order.getState());
            }
        };
        recyclerView_del.setAdapter(adapter);
adapter.startListening();

    }
void init(){
       recyclerView_del= findViewById(R.id.del_order_recyclerview);
layoutManager=new LinearLayoutManager(this);
recyclerView_del.setLayoutManager(layoutManager);
    }
    }
```

Model Class

```
package
com.example.sabzimart.Model;
```

Cart public class Cart {

```
    private String P_name,P_price,P_quantity;
```

```
   Cart(){}

   public Cart(String p_name, String p_price, String p_quantity) {
      P_name = p_name;
      P_price = p_price;
      P_quantity = p_quantity;
   }

   public String getP_name() {
return P_name;
   }

   public void setP_name(String p_name) {
      P_name = p_name;
   }

   public String getP_price() {
return P_price;
   }

   public void setP_price(String p_price) {
      P_price = p_price;
   }

   public String getP_quantity()
{        return P_quantity;
   }

   public void setP_quantity(String p_quantity) {
      P_quantity = p_quantity;
   }
}
```

package
com.example.sabzimart.Model; **Order**

```
public class Order {
   String
Address,Date,House,Name,Product_NAMES,Product_Quantity,State,Time,Phone,Pr
ice;     public Order(){}

           public Order(String address, String date, String house, String name, String
                                                             product_NAMES, String
product_Quantity, String state, String time, String phone, String price) {
      Address = address;
      Date = date;
      House = house;
      Name = name;
```

```
        Product_NAMES = product_NAMES;
        Product_Quantity = product_Quantity;
        State = state;
        Time = time;
        Phone = phone;
        Price = price;
    }

    public String getPrice() {
return Price;
    }

    public void setPrice(String price) {
        Price = price;
    }

    public String getPhone() {
return Phone;
    }

    public void setPhone(String phone) {
        Phone = phone;
    }

    public String getAddress() {
return Address;
    }

    public void setAddress(String address) {
        Address = address;
    }

    public String getDate() {
return Date;
    }

    public void setDate(String date) {
        Date = date;
    }

    public String getHouse() {
return House;
    }

    public void setHouse(String house) {
        House = house;
    }

    public String getName() {
return Name;
    }

    public void setName(String name) {
        Name = name;
    }

    public String
getProduct_NAMES() {         return
Product_NAMES;
    }

    public void setProduct_NAMES(String product_NAMES) {
        Product_NAMES = product_NAMES;
    }

    public String
getProduct_Quantity() {         return
Product_Quantity;
    }

    public void setProduct_Quantity(String product_Quantity) {
        Product_Quantity = product_Quantity;
    }

    public String getState() {
return State;
    }

    public void setState(String state) {
        State = state;
    }

    public String getTime() {
return Time;
   }

    public void setTime(String time) {
        Time = time;
    }
}
```

Products package
com.example.sabzimart.Model;

```
public class Products {
   String
product_name,product_price,image,product_quantity;
Products(){}


   public Products(String product_name, String product_price, String image, String
product_quantity) {        this.product_name = product_name;
this.product_price = product_price;        this.image = image;
this.product_quantity = product_quantity;
   }

   public String
getProduct_quantity() {        return
product_quantity;
   }

   public void setProduct_quantity(String product_quantity) {
this.product_quantity = product_quantity;
   }

   public String getProduct_name()
{        return product_name;
   }

   public void setProduct_name(String product_name) {
this.product_name = product_name;
   }

   public String getProduct_price()
{        return product_price;
   }

   public void setProduct_price(String product_price) {
this.product_price = product_price;
   }

   public String getImage() {
return image;
   }

   public void setImage(String image) {
this.image = image;
   }
}
```

```
package
com.example.sabzimart.Model;
```

Users

```
public class Users {
String name;
   String password;
   String phone;
   Users(){}

   public Users(String name, String password,
String phone) {        this.name = name;
this.password = password;        this.phone =
phone;
   }

   public String getName() {
return name;
   }

   public void setName(String name) {
this.name = name;
   }

   public String getPassword()
{        return password;
   }

   public void setPassword(String password) {
this.password = password;
   }

   public String getPhone() {
return phone;
   }

   public void setPhone(String phone) {
      this.phone = phone;
   }
}
```

ViewHolder

```
public class CartViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener{     public TextView
txtProduct_name,txtProduct_price,txtProduct_quantity;     private
ItemClickedListener itemClickedListener;
```

```
   public CartViewHolder(@NonNull View itemView) {
super(itemView);        txtProduct_name=(TextView)
itemView.findViewById(R.id.cart_product_name);
txtProduct_price=(TextView)itemView.findViewById(R.id.cart_product_p
rice);
txtProduct_quantity=(TextView)itemView.findViewById(R.id.cart_produc
t_quantity);
   }

   @Override     public void onClick(View v) {
itemClickedListener.onClick(v,getAdapterPosition()
,false);

   }
   public void setItemClickedListener(ItemClickedListener itemClickedListener) {
this.itemClickedListener = itemClickedListener;
   }
}

public class OrderViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener{    public TextView
admin_pname,admin_pquantity,admin_Name,admin_time,admin_date,admin_price,
admin_phone,admi n_address,state;

   public void setListener(ItemClickedListener listener) {
this.listener = listener;
   }

   ItemClickedListener listener;

   public OrderViewHolder(@NonNull View itemView) {
super(itemView);
admin_pname=(TextView)itemView.findViewById(R.id.admin_product_
name);
admin_pquantity=(TextView)itemView.findViewById(R.id.admin_produc
t_quantity);
admin_time=(TextView)itemView.findViewById(R.id.time);
admin_date=(TextView)itemView.findViewById(R.id.date);
admin_price=(TextView)itemView.findViewById(R.id.admin_product_pri
ce);
admin_Name=(TextView)itemView.findViewById(R.id.admin_Name);
admin_phone=(TextView)itemView.findViewById(R.id.order_phone_no)
;
admin_address=(TextView)itemView.findViewById(R.id.admin_address
);        state=itemView.findViewById(R.id.admin_state);

   }
```

```
    @Override     public void onClick(View
v) {
listener.onClick(v,getAdapterPosition(),fa
lse);
    }
}


public class ProductViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener{     public TextView productname,productprice,productquantity;
public ImageView productimage;
     public ItemClickedListener product_listener;

    public ProductViewHolder(@NonNull View itemView) {
super(itemView);
productname=(TextView)itemView.findViewById(R.id.product_n
ame);
productprice=(TextView)itemView.findViewById(R.id.product_pri
ce);         productimage=(ImageView)
itemView.findViewById(R.id.product_image);
productquantity=(TextView)itemView.findViewById(R.id.product_
quantity);
    }

    public void setProduct_listener(ItemClickedListener product_listener) {
this.product_listener = product_listener;
    }

    @Override     public void onClick(View v) {
product_listener.onClick(v,getAdapterPosition(),
false);
    }
}
```