

ICP Presentation 2

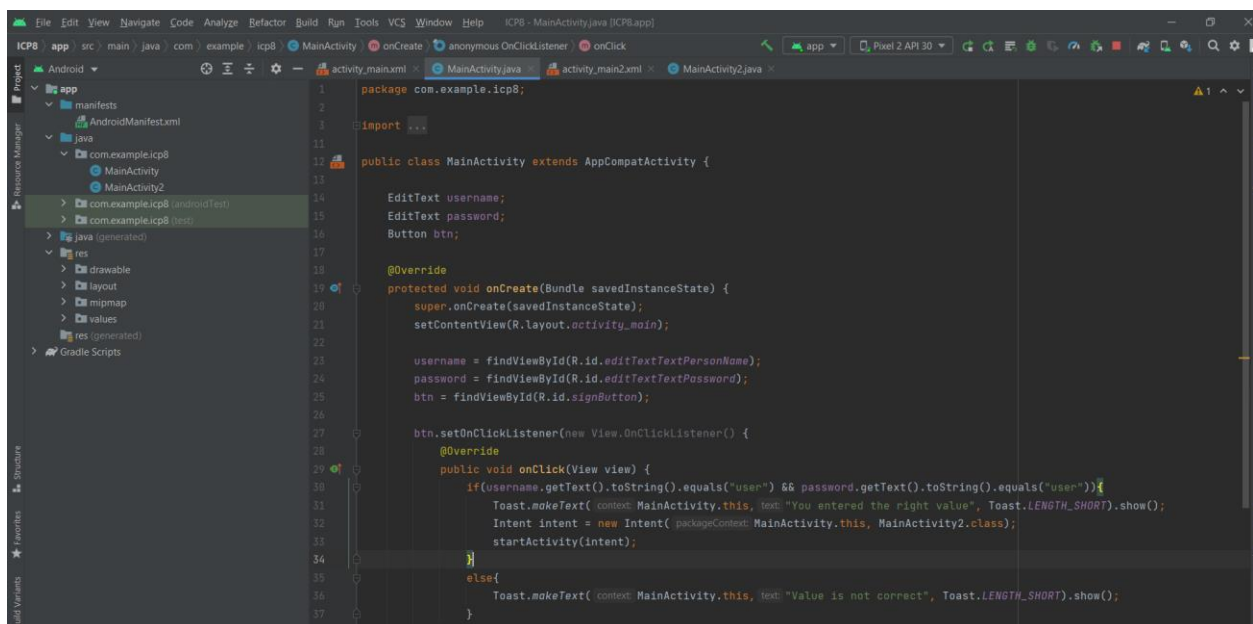
Name: Aqduş Charolia Email: aacy5x@mail.umkc.edu

Name: Affan Charolia Email: aacbb8@mail.umkc.edu

ICP 8

Aim: The goal of this ICP is to create a mobile application, which has the login page and when the user enters login and password as user, they get to the welcome page and when they click on the logout button, they get back to the log in page.

Explanation:



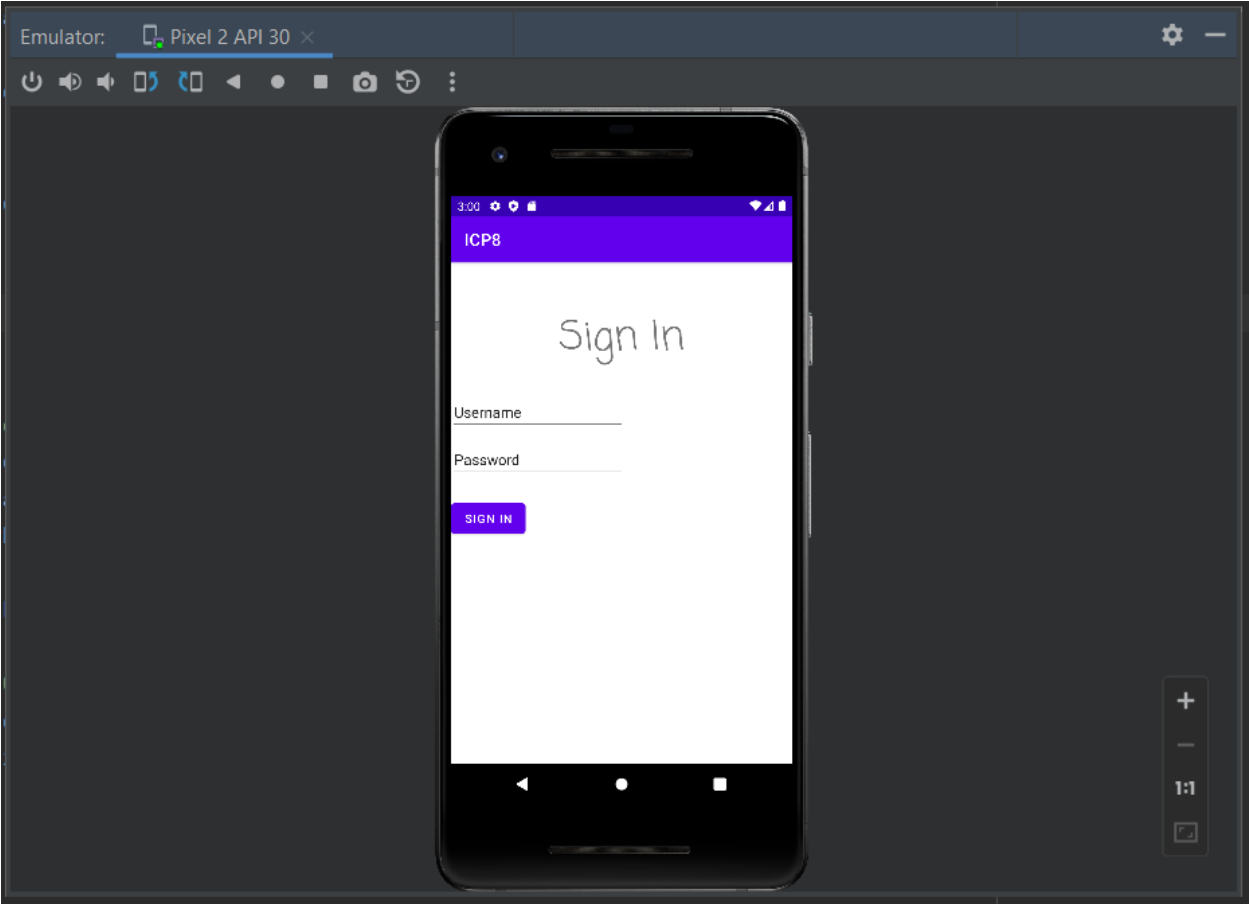
We create the MainActivity.java file to create a view. We create variables and link them to the fields using the id they are being assigned. When the fields are entered and submitted, the strings are matched with the string user and when matched, the toast function displays the title “you are entered the right way”. We then go to the next view using the intent function or if the value is incorrect, we display “value is not correct”.

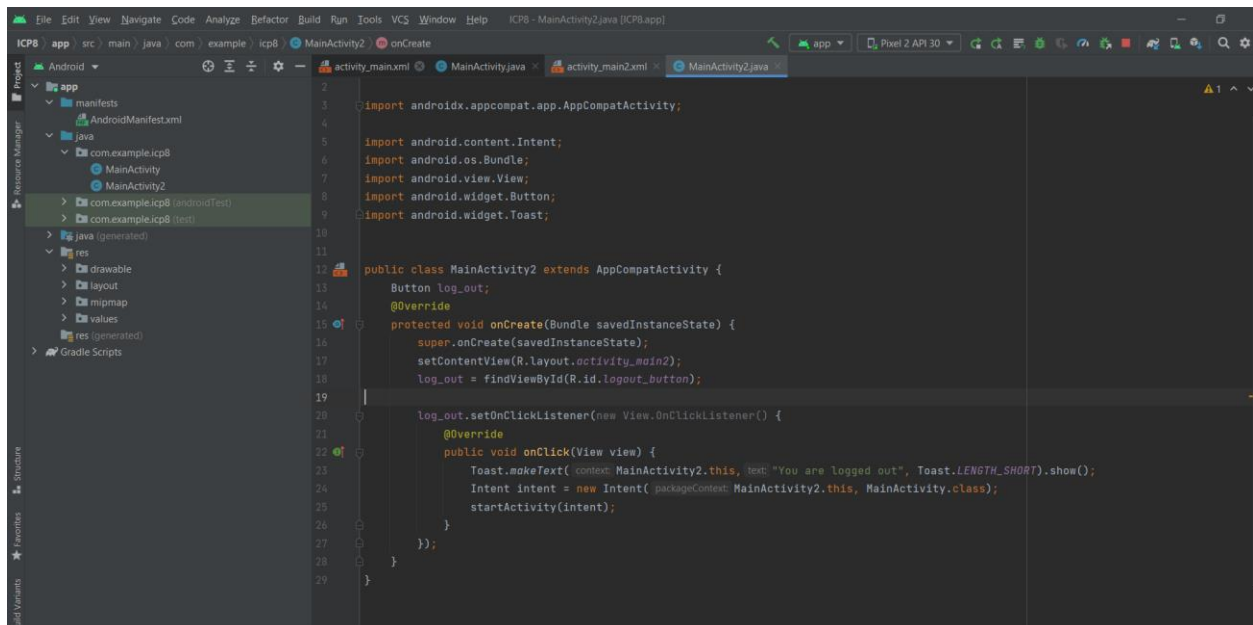
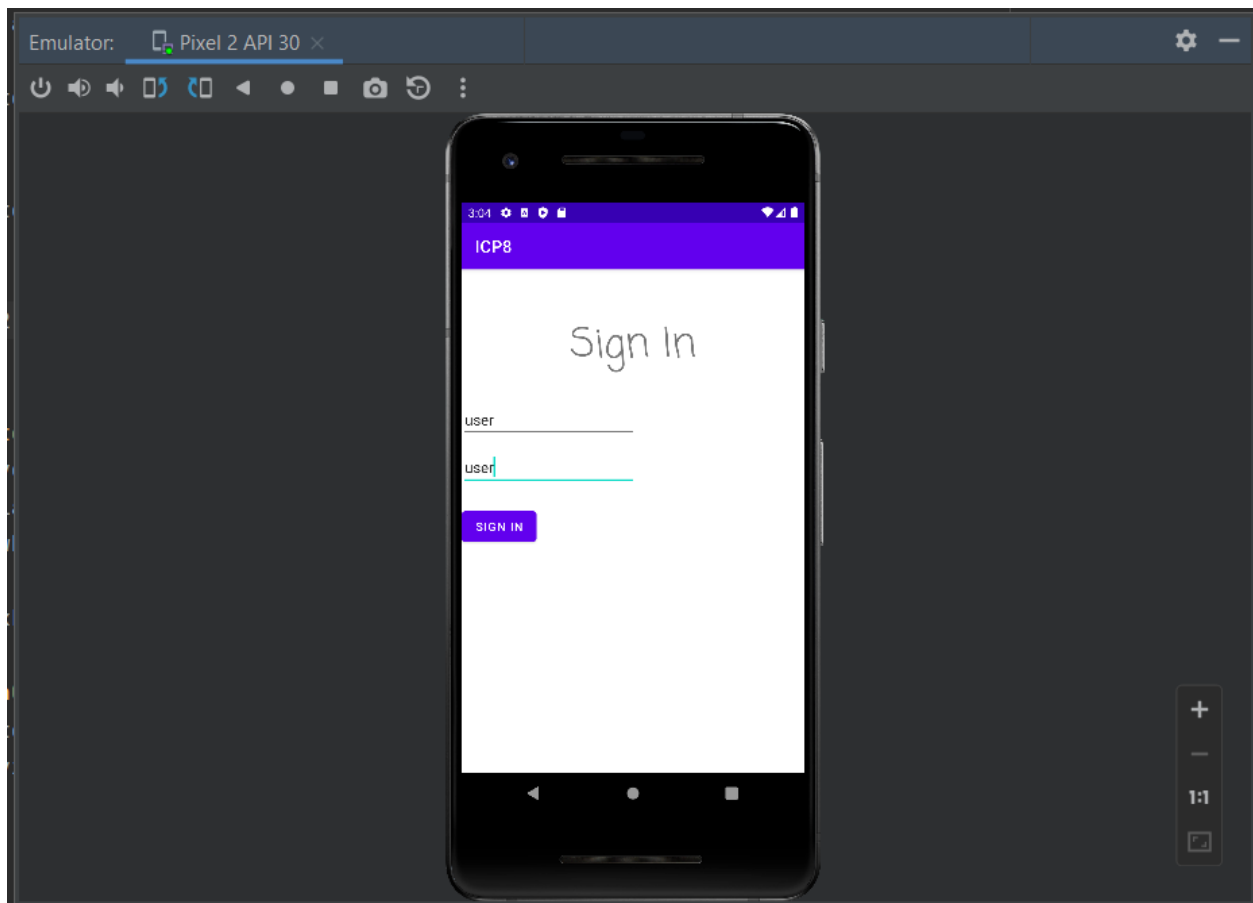
```
activity_main.xml x MainActivity.java x activity_main2.xml x MainActivity2.java x
Code Split Design
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <TextView
10     android:id="@+id/logo"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     android:fontFamily="casual"
14     android:text="@string/sign_in"
15     android:textSize="48sp"
16     app:layout_constraintBottom_toBottomOf="parent"
17     app:layout_constraintLeft_toLeftOf="parent"
18     app:layout_constraintRight_toRightOf="parent"
19     app:layout_constraintTop_toTopOf="parent"
20     app:layout_constraintVertical_bias="0.103" />
21
22   <Button
23     android:id="@+id/signButton"
24     android:layout_width="wrap_content"
25     android:layout_height="wrap_content"
26     android:layout_marginTop="24dp"
27     android:text="@string/sign_in"
28     app:layout_constraintTop_toBottomOf="@id/editTextTextPassword"

```

```
ain.xml
activity_main.xml x MainActivity.java x activity_main2.xml x MainActivity2.java x
Code Split Design
29   tools:ignore="MissingConstraints"
30   tools:layout_editor_absoluteX="160dp" />
31
32   <EditText
33     android:id="@+id/editTextTextPersonName"
34     android:layout_width="wrap_content"
35     android:layout_height="wrap_content"
36     android:layout_marginTop="30dp"
37     android:ems="10"
38     android:inputType="textPersonName"
39     android:text="Username"
40     app:layout_constraintTop_toBottomOf="@id/logo"
41     tools:layout_editor_absoluteX="103dp" />
42
43   <EditText
44     android:id="@+id/editTextTextPassword"
45     android:layout_width="wrap_content"
46     android:layout_height="wrap_content"
47     android:layout_marginTop="12dp"
48     android:ems="10"
49     android:inputType="textPersonName"
50     android:text="Password"
51     app:layout_constraintTop_toBottomOf="@id/editTextTextPersonName"
52     tools:ignore="TouchTargetSizeCheck"
53     tools:layout_editor_absoluteX="97dp" />
54
55 </androidx.constraintlayout.widget.ConstraintLayout>

```

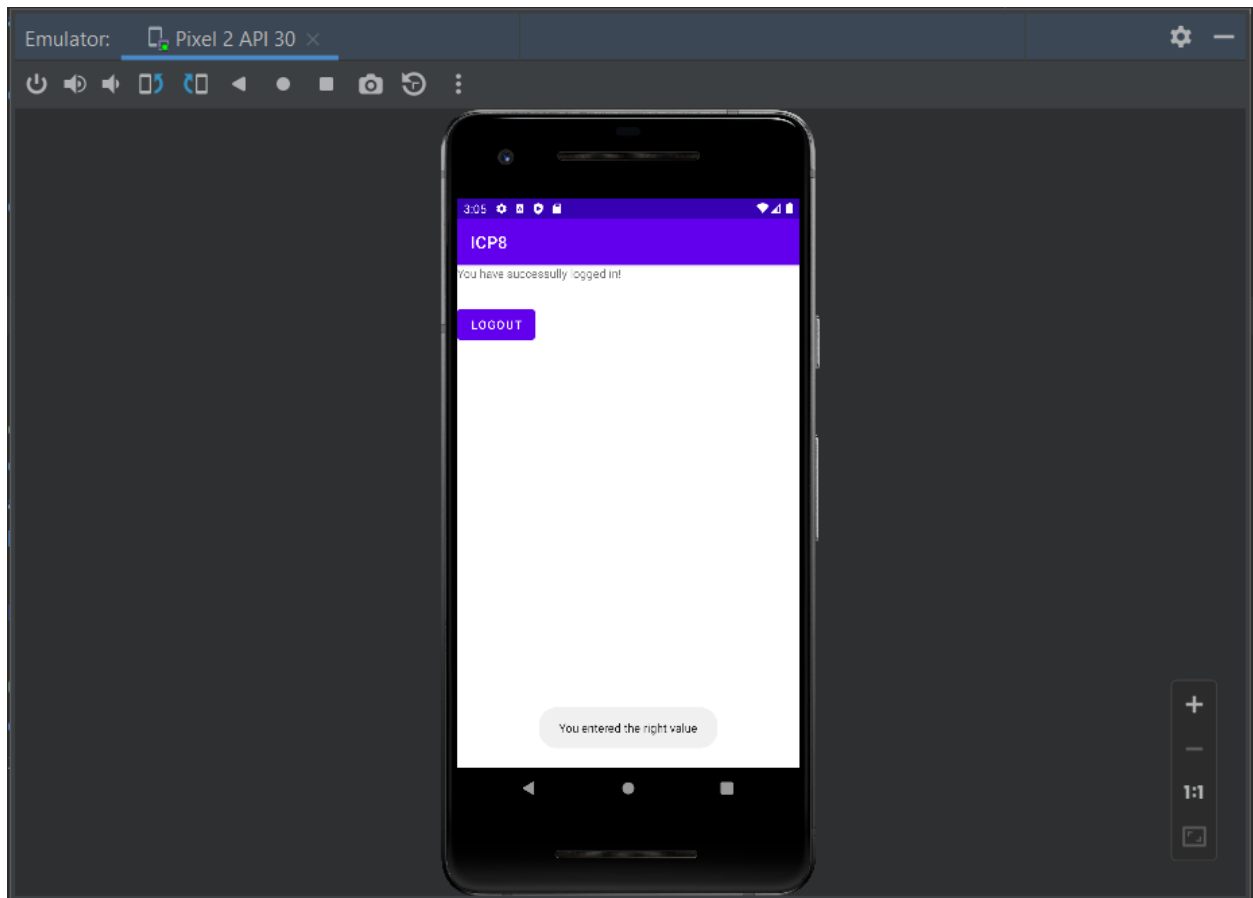


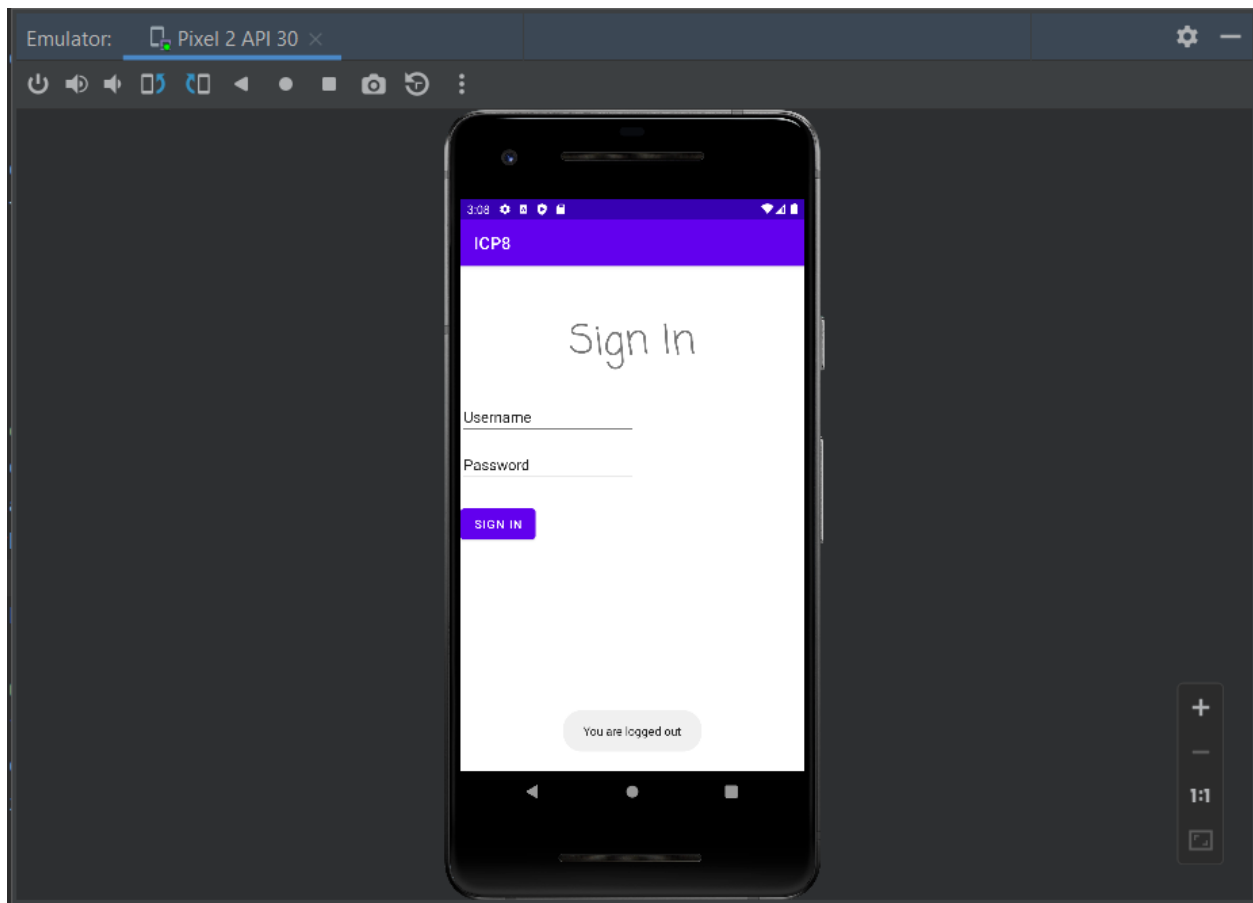


When we get to the welcome page, it displays the string and the logout button. On clicking the logout button we get to the login page again, along with the displayed “You are logged out” string.

```
activity_main.xml x MainActivity.java x activity_main2.xml x MainActivity2.java x
Code Split Design

1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity2">
8
9   <TextView
10     android:id="@+id/textView1"
11     android:layout_width="210dp"
12     android:layout_height="19dp"
13     android:text="You have successfully logged in!"
14     tools:layout_editor_absoluteX="7dp"
15     tools:layout_editor_absoluteY="144dp" />
16
17   <Button
18     android:id="@+id/logout_button"
19     android:layout_width="wrap_content"
20     android:layout_height="wrap_content"
21     android:layout_marginTop="28dp"
22     android:text="Logout"
23     app:layout_constraintTop_toBottomOf="@id/textView1"
24     tools:layout_editor_absoluteX="64dp" />
25 </androidx.constraintlayout.widget.ConstraintLayout>
```





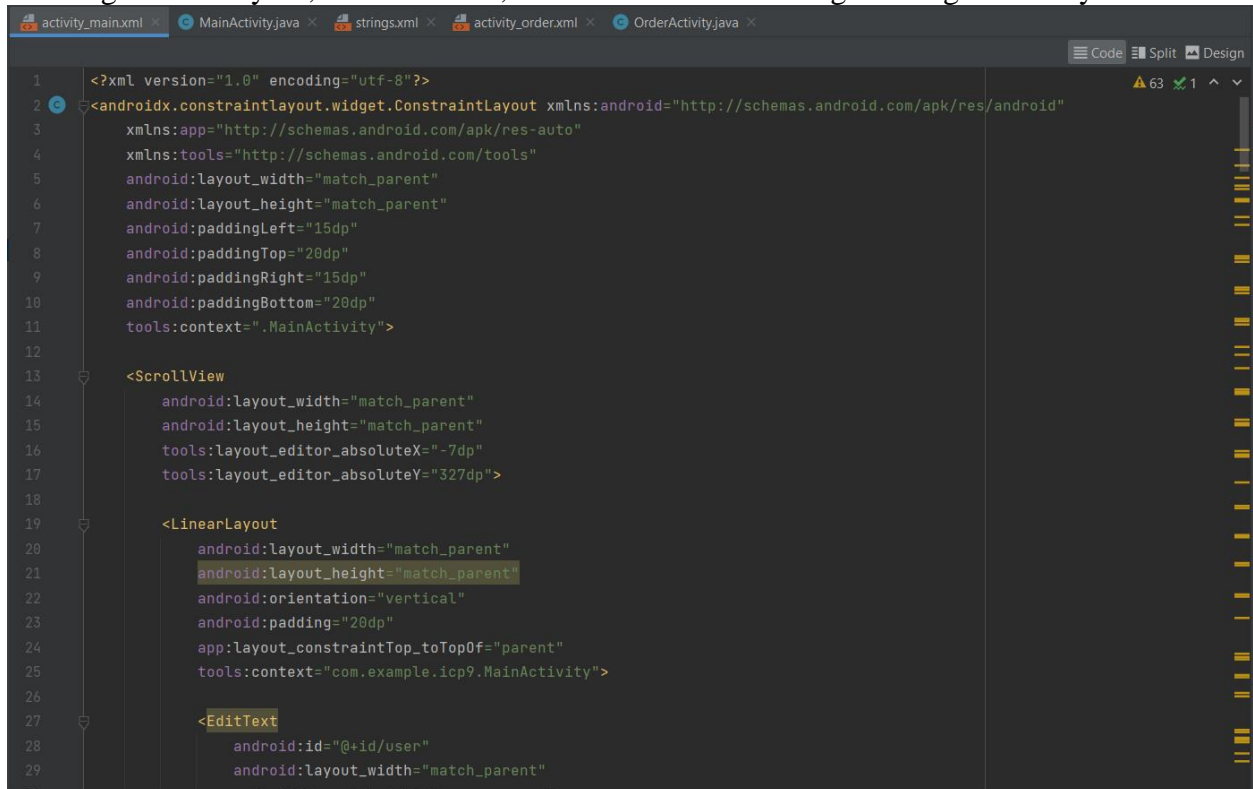
Conclusion: We were able to create a login page which took in the inputs and when successful credentials were entered it took us to the welcome page with correct message being displayed on success and error. Further, on clicking the logout button we were able to go back to the login page with correct message being displayed.

ICP 9

Aim: The goal of this ICP is to create a pizza ordering mobile application which will allow the user to make some selection of their choice and they will be able to see their order summary and place the order which will get sent to them by mail.

Explanation:

Creating a linear layout, will a scroller, so that all the items will get arranged linearly.



```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:paddingLeft="15dp"
8      android:paddingTop="20dp"
9      android:paddingRight="15dp"
10     android:paddingBottom="20dp"
11     tools:context=".MainActivity">
12
13     <ScrollView
14         android:layout_width="match_parent"
15         android:layout_height="match_parent"
16         tools:layout_editor_absoluteX="-7dp"
17         tools:layout_editor_absoluteY="327dp">
18
19         <LinearLayout
20             android:layout_width="match_parent"
21             android:layout_height="match_parent"
22             android:orientation="vertical"
23             android:padding="20dp"
24             app:layout_constraintTop_toTopOf="parent"
25             tools:context="com.example.icp9.MainActivity">
26
27             <EditText
28                 android:id="@+id/user"
29                 android:layout_width="match_parent"
```

Taking input for username and email address.

```
27 <EditText
28     android:id="@+id/user"
29     android:layout_width="match_parent"
30     android:layout_height="wrap_content"
31     android:layout_marginBottom="16sp"
32     android:hint="Your Name"
33     android:inputType="textCapWords" />
34
35 <EditText
36     android:id="@+id/email"
37     android:layout_width="match_parent"
38     android:layout_height="wrap_content"
39     android:layout_marginBottom="16sp"
40     android:hint="Your Email address"
41     android:inputType="textCapWords" />
42
43
44 <TextView
45     android:layout_width="wrap_content"
46     android:layout_height="wrap_content"
47     android:text="Select Pizza Size"
48     android:textStyle="bold" />
49
50 <RadioGroup xmlns:android="http://schemas.android.com/apk/res/android"
51     android:id="@+id/size"
52     android:layout_width="match_parent"
53     android:layout_height="wrap_content"
54     android:orientation="vertical">
```

Selecting the size of pizza

```
56 <RadioButton
57     android:id="@+id/size_small"
58     android:layout_width="wrap_content"
59     android:layout_height="wrap_content"
60     android:layout_marginBottom="16sp"
61     android:buttonTint="#708090"
62     android:onClick="onSizeSelected"
63     android:paddingLeft="24dp"
64     android:text="Small"
65     android:textColor="@android:color/holo_blue_dark"
66     android:textSize="16sp" />
67
68 <RadioButton
69     android:id="@+id/size_medium"
70     android:layout_width="wrap_content"
71     android:layout_height="wrap_content"
72     android:layout_marginBottom="16sp"
73     android:buttonTint="#708090"
74     android:onClick="onSizeSelected"
75     android:paddingLeft="24dp"
76     android:text="Medium"
77     android:textColor="@android:color/holo_blue_dark"
78     android:textSize="16sp" />
79
80 <RadioButton
81     android:id="@+id/size_large"
82     android:layout_width="wrap_content"
83     android:layout_height="wrap_content"
84     android:layout_marginBottom="16sp"
85     android:buttonTint="#708090"
86     android:onClick="onSizeSelected"
87     android:paddingLeft="24dp"
88     android:text="Large"
89     android:textColor="@android:color/holo_blue_dark"
90     android:textSize="16sp" />
```

Select the crust of the pizza according to your choice.


```
79
80
81     <RadioButton
82         android:id="@+id/size_large"
83         android:layout_width="wrap_content"
84         android:layout_height="wrap_content"
85         android:layout_marginBottom="16sp"
86         android:buttonTint="#708090"
87         android:onClick="onSizeSelected"
88         android:paddingLeft="24dp"
89         android:text="Large"
90         android:textColor="@android:color/ho_lo_blue_dark"
91         android:textSize="16sp" />
92 </RadioGroup>
93
94 <TextView
95     android:layout_width="wrap_content"
96     android:layout_height="wrap_content"
97     android:text="Select Crust"
98     android:textStyle="bold" />
99
100 <RadioGroup xmlns:android="http://schemas.android.com/apk/res/android"
101     android:id="@+id/crust"
102     android:layout_width="match_parent"
103     android:layout_height="wrap_content"
104     android:orientation="vertical"
105     android:text="Select Crust">
106
107     <RadioButton
108         android:id="@+id/originalPan"
```

```
<RadioButton
    android:id="@+id/originalPan"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16sp"
    android:buttonTint="#708090"
    android:onClick="onCrustSelected"
    android:paddingLeft="24dp"
    android:text="Original Pan Pizza"
    android:textColor="@android:color/ho_lo_blue_dark"
    android:textSize="16sp" />

<RadioButton
    android:id="@+id/handmadePan"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16sp"
    android:buttonTint="#708090"
    android:onClick="onCrustSelected"
    android:paddingLeft="24dp"
    android:text="Handmade Pan"
    android:textColor="@android:color/ho_lo_blue_dark"
    android:textSize="16sp" />

<RadioButton
    android:id="@+id/thinNCrispy"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16sp"
    android:buttonTint="#708090"
    android:onClick="onCrustSelected"
    android:paddingLeft="24dp"
    android:text="Thin N Crispy"
    android:textColor="@android:color/ho_lo_blue_dark"
    android:textSize="16sp" />

</RadioGroup>
```

Selecting the topping. Since it is a checkbox, you can select multiple options.

```
<RadioButton
    android:id="@+id/thinNCrispy"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16sp"
    android:buttonTint="#708090"
    android:onClick="onCrustSelected"
    android:paddingLeft="24dp"
    android:text="Thin'N' Crispy"
    android:textColor="@android:color/ho_lo_blue_dark"
    android:textSize="16sp" />

</RadioGroup>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Toppings"
    android:textStyle="bold" />

<CheckBox
    android:id="@+id/mushrooms"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16sp"
    android:buttonTint="#708090"
    android:paddingLeft="24dp"
    android:text="Mushrooms"
```

```
    android:textColor="@android:color/ho_lo_blue_dark"
    android:textSize="16sp" />

    <CheckBox
        android:id="@+id/pineapple"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16sp"
        android:buttonTint="#708090"
        android:paddingLeft="24dp"
        android:text="Pineapple"
        android:textColor="@android:color/ho_lo_blue_dark"
        android:textSize="16sp" />

    <CheckBox
        android:id="@+id/cheese"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16sp"
        android:buttonTint="#708090"
        android:paddingLeft="24dp"
        android:text="Cheese"
        android:textColor="@android:color/ho_lo_blue_dark"
        android:textSize="16sp" />

    <CheckBox
        android:id="@+id/chicken"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

The user can select the quantity of pizza they want to order. They can increase or decrease the quantity using the buttons.

```
        android:id="@+id/chicken"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16sp"
        android:buttonTint="#708090"
        android:paddingLeft="24dp"
        android:text="Chicken"
        android:textColor="@android:color/ho_lo_blue_dark"
        android:textSize="16sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Quantity"
    android:textStyle="bold" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:paddingLeft="10dp"
    android:paddingRight="10dp">

    <Button
        style="@style/Widget.AppCompat.Button.Colored"
        android:layout_width="48dp"
        android:layout_height="48dp"
        android:layout_marginRight="8dp"
        android:onClick="decrement" />

        <Button
            style="@style/Widget.AppCompat.Button.Colored"
            android:layout_width="48dp"
            android:layout_height="48dp"
            android:layout_marginRight="8dp"
            android:onClick="increment" />

    <TextView
        android:id="@+id/tv_quantity"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginRight="8dp"
        android:text="1" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:orientation="horizontal">

    <Button
        android:id="@+id/btn_order"
        style="@style/Widget.AppCompat.Button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Order" />

    <Button
        android:id="@+id/btn_cancel"
        style="@style/Widget.AppCompat.Button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Cancel" />

</LinearLayout>
```

The user has two options to select. Either he can view the order summary or submit the order and receive the confirmation in the mail.

```
        android:layout_height="48dp"
        android:layout_marginRight="8dp"
        android:onClick="decrement"
        android:text="-" />

<TextView
    android:id="@+id/tv_quantity"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginRight="8dp"
    android:text="1" />

<Button
    style="@style/Widget.AppCompat.Button.Colored"
    android:layout_width="48dp"
    android:layout_height="48dp"
    android:onClick="increment"
    android:text="+" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:orientation="horizontal">

    <Button
        android:id="@+id/btn_order"
        style="@style/Widget.AppCompat.Button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Order" />

    <Button
        android:id="@+id/btn_cancel"
        style="@style/Widget.AppCompat.Button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Cancel" />

</LinearLayout>
```

```

        <Button
            android:id="@+id/btn_order"
            style="@style/Widget.AppCompat.Button"
            android:layout_width="wrap_content"
            android:layout_height="50dp"
            android:layout_marginRight="8dp"
            android:text="Order"
            android:textColor="#FFFFFF" />
    */
    <Button
        android:id="@+id/btn_summary"
        android:layout_width="wrap_content"
        android:layout_height="50dp"
        android:onClick="submitOrder"
        android:paddingLeft="10dp"
        android:paddingRight="10dp"
        android:text="View Order Summary"
        android:textColor="#FFFFFF" />
    </LinearLayout>
</LinearLayout>
</ScrollView>
</androidx.constraintlayout.widget.ConstraintLayout>

```

Assigning all the variables based on their configuration, like buttons, checkbox and radio button.

```

1  package com.example.icp9;
2
3  import ...
17
18  public class MainActivity extends AppCompatActivity {
19      private static final String MAIN_ACTIVITY_TAG = "MainActivity";
20
21
22      EditText userNameView;
23      EditText userEmailView;
24      CheckBox cheese;
25      CheckBox mushrooms;
26      CheckBox chicken;
27      CheckBox pineapple;
28      Button OrderBtn;
29      final int PIZZA_PRICE = 10;
30      final int CHEESE_PRICE = 3;
31      final int MUSHROOMS_PRICE = 2;
32      final int CHICKEN_PRICE = 5;
33      final int PINEAPPLE_PRICE = 1;
34      String size;
35      String crust;
36      int quantity = 1;
37
38      @Override
39      protected void onCreate(Bundle savedInstanceState) {
40          super.onCreate(savedInstanceState);
41          setContentView(R.layout.activity_main);
42          userNameView = findViewById(R.id.user);
43          userEmailView = findViewById(R.id.email);
44

```

When order button is clicked, the send mail function is called.

```
pineapple = findViewById(R.id.pineapple);
cheese = findViewById(R.id.cheese);
mushrooms = findViewById(R.id.mushrooms);
chicken = findViewById(R.id.chicken);
OrderBtn = findViewById(R.id.btn_order);

OrderBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        sendEmail();
    }
});

}

@
public void onSizeSelected(View view){
    boolean checked = ((RadioButton) view).isChecked();
    // Check which radio button was clicked
    switch(view.getId()) {
        case R.id.size_small:
            if (checked)
                size = "Small";
            break;
        case R.id.size_medium:
            if (checked)
                size = "Medium";
            break;
        case R.id.size_large:
            if (checked)
                size = "Large";
            break;
    }
}
```

OnCrustSelected function is used to manage the radiobutton for crust selection.

```
@
public void onCrustSelected(View view){
    boolean checked = ((RadioButton) view).isChecked();

    // Check which radio button was clicked
    switch(view.getId()) {
        case R.id.originalPan:
            if (checked)
                crust = "Original Pan Pizza";
            break;
        case R.id.handmadePan:
            if (checked)
                crust = "Handmade Pan";
            break;
        case R.id.thinNCrispy:
            if (checked)
                crust = "Thin'N Crispy";
            break;
    }
}

public void sendEmail() {

    // read user input
    String userName = userNameView.getText().toString();
    // read user email
    String userEmail = userEmailView.getText().toString();
    String[] userEmails = userEmail.split( regex: " " );

    // check if pineapple is selected
    boolean hasPineapple = pineapple.isChecked();
}
```

The send email function takes all the ingredients and create the message string using order summary message function and calculates the total amount of the bill and sends the mail to the user using the email id given by the user and self-generated subject.

```
09     boolean hasPineapple = pineapple.isChecked();
10     // check if cheese is selected
11     boolean hasCheese = cheese.isChecked();
12     // check if mushroom is selected
13     boolean hasMushrooms = mushrooms.isChecked();
14     // check if chicken is selected
15     boolean hasChicken = chicken.isChecked();
16
17
18     // calculate and store the total price
19     float totalPrice = calculatePrice(hasCheese, hasMushrooms, hasChicken, hasPineapple);
20
21     // create and store the order summary details
22     String orderSummaryMessage = createOrderSummary(userName, hasCheese, hasMushrooms, hasChicken, hasPineapple, totalPrice);
23
24     Intent emailIntent = new Intent(Intent.ACTION_SEND);
25     Log.i("tag: Send_email", "msg:");
26     emailIntent.setData(Uri.parse("mailto:"));
27     emailIntent.setType("text/plain");
28     emailIntent.putExtra(Intent.EXTRA_EMAIL, userEmails);
29     emailIntent.putExtra(Intent.EXTRA_SUBJECT, "Pizza Delivery Details");
30     emailIntent.putExtra(Intent.EXTRA_TEXT, orderSummaryMessage);
31     try {
32         startActivity(Intent.createChooser(emailIntent, "Choose any email client"));
33         finish();
34     } catch (android.content.ActivityNotFoundException ex) {
35         Toast.makeText(context: MainActivity.this, text: "There is no email client installed.", Toast.LENGTH_SHORT).show();
36     }
37 }
```

Submit order calculates the price for the pizza.

```
public void submitOrder(View view) {

    String userName = userNameView.getText().toString();

    boolean hasPineapple = pineapple.isChecked();

    boolean hasCheese = cheese.isChecked();

    boolean hasMushrooms = mushrooms.isChecked();

    boolean hasChicken = chicken.isChecked();

    float totalPrice = calculatePrice(hasCheese, hasMushrooms, hasChicken, hasPineapple);

    String orderSummaryMessage = createOrderSummary(userName, hasCheese, hasMushrooms, hasChicken, hasPineapple, totalPrice);

    Intent redirect = new Intent(context: MainActivity.this, OrderActivity.class);
    redirect.putExtra(name: "MESSAGE", orderSummaryMessage);
    MainActivity.this.startActivity(redirect);
}

private String boolToString(boolean bool) {
    return bool ? (getString(R.string.yes)) : (getString(R.string.no));
}

// Order Summary Details
private String createOrderSummary(String userName, boolean hasCheese, boolean hasMushrooms, boolean hasChicken, boolean hasPineapple) {
    String orderSummaryMessage = getString(R.string.order_summary_name, userName) + "\n" +
        getString(R.string.order_details) + "\n" +
        getString(R.string.order_summary_pineapple, boolToString(hasPineapple)) + "\n" +
        getString(R.string.order_summary_cheese, boolToString(hasCheese)) + "\n" +
```

```

private String createOrderSummary(String userName, boolean hasCheese, boolean hasMushrooms, boolean hasChicken, boolean hasPineapple) {
    String orderSummaryMessage = getString(R.string.order_summary_name, userName) + "\n" +
        getString(R.string.order_details) + "\n" +
        getString(R.string.order_summary_pineapple, boolToString(hasPineapple)) + "\n" +
        getString(R.string.order_summary_cheese, boolToString(hasCheese)) + "\n" +
        getString(R.string.order_summary_mushrooms, boolToString(hasMushrooms)) + "\n" +
        getString(R.string.order_summary_chicken, boolToString(hasChicken)) + "\n" +
        "Crust?" + crust + "\n" +
        "Size?" + size + "\n" +
        getString(R.string.order_summary_quantity, quantity) + "\n" +
        getString(R.string.order_summary_total_price, price) + "\n" +
        getString(R.string.thank_you);

    return orderSummaryMessage;
}

/**
 * Method to calculate the total price
 *
 * @return total Price
 */
private float calculatePrice(boolean hasCheese, boolean hasMushrooms, boolean hasChicken, boolean hasPineapple) {
    int basePrice = PIZZA_PRICE;
    if (hasCheese) {
        basePrice += CHEESE_PRICE;
    }
    if (hasMushrooms) {
        basePrice += MUSHROOMS_PRICE;
    }
    if (hasChicken) {
        basePrice += CHICKEN_PRICE;
    }
    if (hasPineapple) {

```

Increment and decrement function is used to increase or decrease the quantity of pizza.

```

        if (hasPineapple) {
            basePrice += PINEAPPLE_PRICE;
        }
        return quantity * basePrice;
    }

    /**
     * This method displays the given quantity value on the screen.
     */
    private void display(int number) {
        TextView quantityTextView = (TextView) findViewById(R.id.tv_quantity);
        quantityTextView.setText("" + number);
    }

    /**
     * This method increments the quantity of coffee cups by one
     *
     * @param view on passes the view that we are working with to the method
     */
    public void increment(View view) {
        if (quantity < 100) {
            quantity = quantity + 1;
            display(quantity);
        } else {
            Log.i(tag: "MainActivity", msg: "Please select less than one hundred pizza");
            Context context = getApplicationContext();
            String lowerLimitToast = getString(R.string.too_much_coffee);
            int duration = Toast.LENGTH_SHORT;
            Toast toast = Toast.makeText(context, lowerLimitToast, duration);

```

```

        Log.i(tag: "MainActivity", msg: "Please select less than one hundred pizza");
        Context context = getApplicationContext();
        String lowerLimitToast = getString(R.string.too_much_coffee);
        int duration = Toast.LENGTH_SHORT;
        Toast toast = Toast.makeText(context, lowerLimitToast, duration);
        toast.show();
        return;
    }

    public void decrement(View view) {
        if (quantity > 1) {
            quantity = quantity - 1;
            display(quantity);
        } else {
            Log.i(tag: "MainActivity", msg: "Please select atleast one pizza");
            Context context = getApplicationContext();
            String upperLimitToast = getString(R.string.too_little_coffee);
            int duration = Toast.LENGTH_SHORT;
            Toast toast = Toast.makeText(context, upperLimitToast, duration);
            toast.show();
            return;
        }
    }
}

```

This is the new activity page which shows the order summary and a button to go back to the main page.

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/order"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".OrderActivity">

    <TextView
        android:id="@+id/Ordertxt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="74dp"
        android:text="Order Summary"
        android:textSize="25sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/Details"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:text="Details"
        android:textSize="20sp"

```



```
android:layout_height="wrap_content"
android:layout_marginTop="15dp"
android:text="Details"
android:textSize="20sp"
android:textStyle="italic"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/Ordertxt" />
```

<Button

```
android:id="@+id/BackToOrder"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="32dp"
android:text="Go to Order"
android:padding="15dp"
android:onClick="onClick"
android:textColor="#FFFFFF"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.181"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/Details" />
```

</androidx.constraintlayout.widget.ConstraintLayout>

This page displays the order summary and on clicking the go to the order button, goes back to the main page.

```
package com.example.icp9;

import androidx.appcompat.app.AppCompatActivity;

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

public class OrderActivity extends AppCompatActivity {
    TextView Details;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_order);
        Details = (TextView)findViewById(R.id.Details);

        Details.setText(getIntent().getStringExtra( name: "MESSAGE"));
    }
    public void onClick(View view) {
        finish();
    }
}
```

Adding the name and email id, and selecting the choice of pizza you want to have.

5:07 61%

ICP9

Affan

affan Charolia.ac@gmail.com

Select Pizza Size

☒ Small

☐ Medium

☐ Large

Select Crust

☒ Original Deep Dish

1 2 3 4 5 6 7 8 9 0

q w e r t y u i o p

a s d f g h j k l

↑ z x c v b n m ↵

?123 , . English ✓

▼

5:07



Bluetooth LTE 61%

ICP9

☐ Medium

☐ Large

Select Crust

☒ Original Pan Pizza

☐ Handmade Pan

☐ Thin'N Crispy

Toppings

☐ Mushrooms

☒ Pineapple

☒ Cheese

☒ Chicken

Quantity



2



ORDER

VIEW ORDER SUMMARY

ICP9

Order Summary

Dear Affan,

Your order details:

Add Pineapple? Yes

Add Cheese? Yes

Add Mushrooms? No

Add Chicken? Yes

Crust? Original Pan Pizza

Size? Small

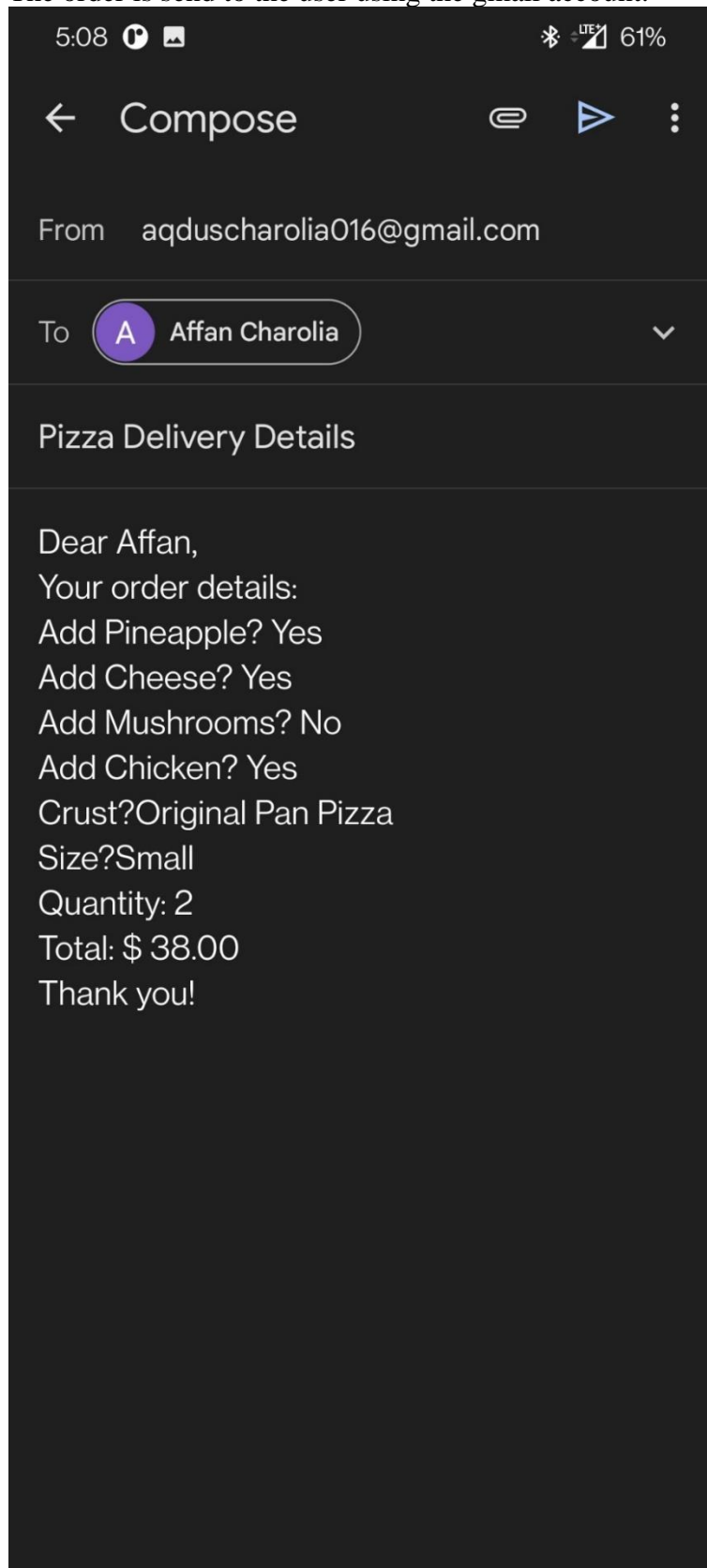
Quantity: 2

Total: \$ 38.00

Thank you!

GO TO ORDER

The order is send to the user using the gmail account.



Conclusion: We were able to create pizza ordering application where you can select the things you need on it and the quantity of pizza. Then the selections were displayed on the summary page along with the cost. The order placed is then sent to the user by email.