Name : H.A.P.C.J.Jayathilaka

Indexed No : 17020387

Date : 07/06/2020

Develop a User Interface for an Online Banking Application

Implemented the restrictions for inputs

- **National ID / Passport Number(max 20 characters)**

    Max 20 characters are restricted in activity_main.xml as follows;

    ```xml
    <EditText
        android:id="@+id/nationalid"
        android:layout_width="393dp"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:hint="National ID / Passport Number"
        android:inputType="textFilter"
        android:maxLength="20"
        android:singleLine="true"/>
    ```

- **Account Number(max 20 digits, first char non zero)**

    Max 20 digits are restricted in activity_main.xml as follows;

    ```xml
    <EditText
        android:id="@+id/accountnumber"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:hint="Account Number"
        android:inputType="number"
        android:maxLength="20"
    />
    ```

First char non zero are restricted as follows in Main Activity

```java
//first element should not be zero
e2.addTextChangedListener(new TextWatcher(){
    public void onTextChanged(CharSequence s, int start, int before, int count)
    {
        if (e2.getText().toString().matches( regex: "^0") )
        {
            // Not allowed

            Toast.makeText(getApplicationContext(), text: "NOT ALLOWED", Toast.LENGTH_LONG).show();
            e2.setText("");
        }
    }
    @Override
    public void afterTextChanged(Editable arg0) { }
    @Override
    public void beforeTextChanged(CharSequence s, int start, int count, int after) {}
});
```

Screen Output :

When entering first value as zero to the account number it give "not allowed"message
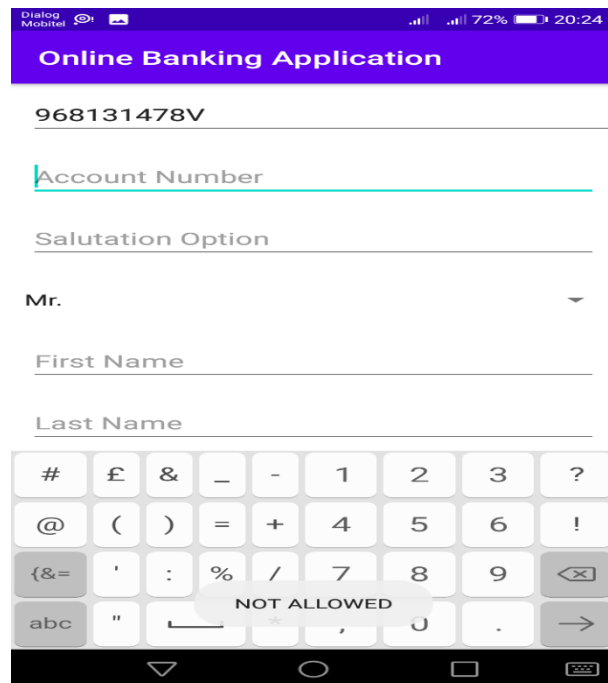
- **Salutation option (Mr./Mrs./Prof./Dr./Rev./Other) and if user select other (10 Alpha)**

  Code is written in in activity_main.xml as follows;

```xml
<EditText
    android:id="@+id/salutation"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="Salutation Option"
    android:digits="abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
    android:inputType="text"
    android:maxLength="10"
    />

<Spinner
    android:id="@+id/spinner"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Salutation Option"

    />
```

In Main Activity we set the spinner to edit text value and make Others user text to alphabetic max-length 10 characters

```java
spinner = (Spinner) findViewById(R.id.spinner);
text = (EditText) findViewById(R.id.salutation);

list = new ArrayList<String>();
list.add("Mr.");
list.add("Ms.");
list.add("Prof.");
list.add("Dr.");
list.add("Rev.");
list.add("Other.");




ArrayAdapter<String> adp = new ArrayAdapter<String>
        ( context: this, android.R.layout.simple_dropdown_item_1line, list);
adp.setDropDownViewResource(android.R.layout.simple_dropdown_item_1line);

spinner.setAdapter(adp);

spinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

    @Override
    public void onItemSelected(AdapterView<?> arg0, View arg1,
                               int arg2, long arg3) {
        // TODO Auto-generated method stub


    }
});
```
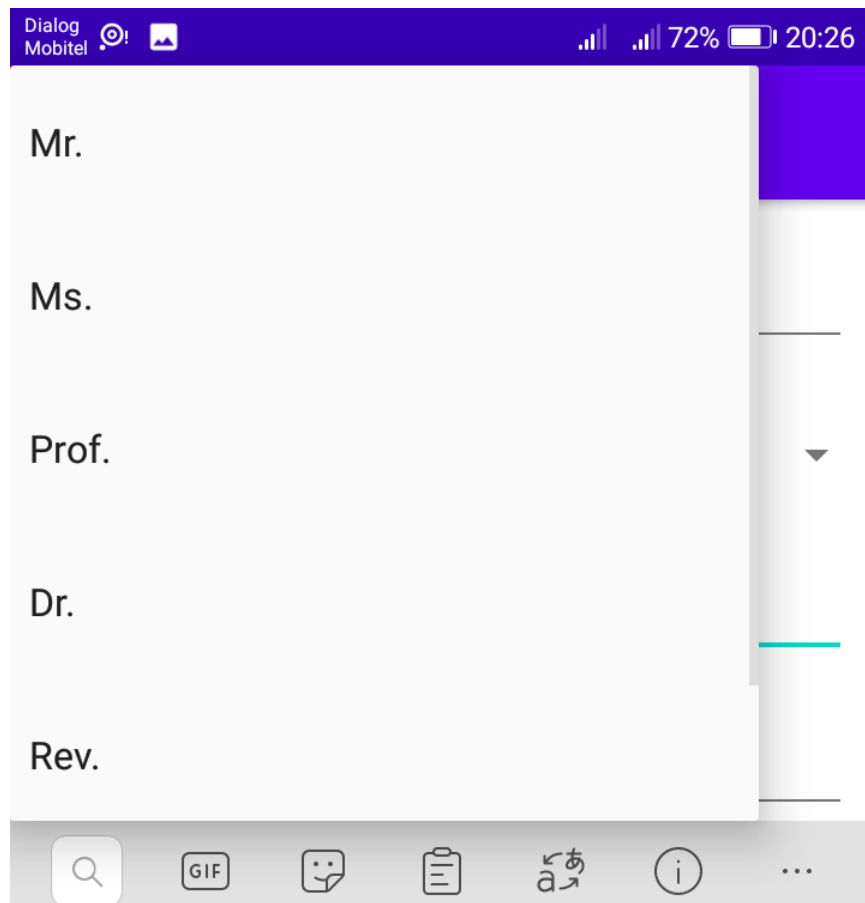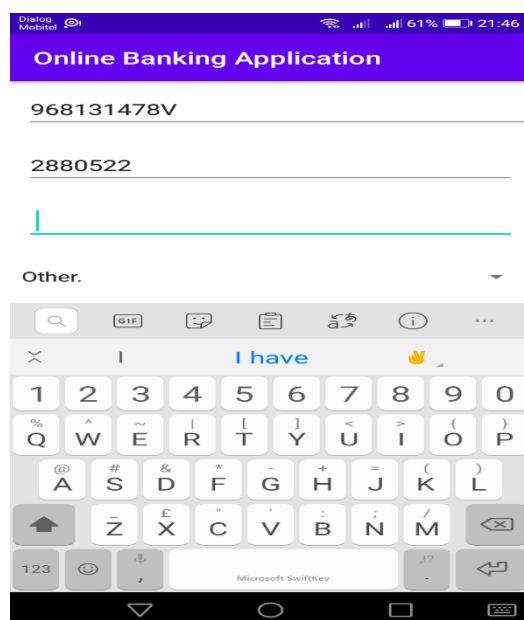
3

Screen Output:

When "Other" is selected as Salutation, you can user defined it

- **First Name(Alpha.50 characters)**

```xml
<EditText
    android:id="@+id/firstname"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="First Name"
    android:digits="abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
    android:inputType="textPersonName"
    android:maxLength="50"
    />
```

- **Last Name(Alpha.50 characters)**

```xml
<EditText
    android:id="@+id/lastname"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:maxLength="50"
    android:digits="abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
    android:hint="Last Name"
    android:inputType="textPersonName"
    />
```

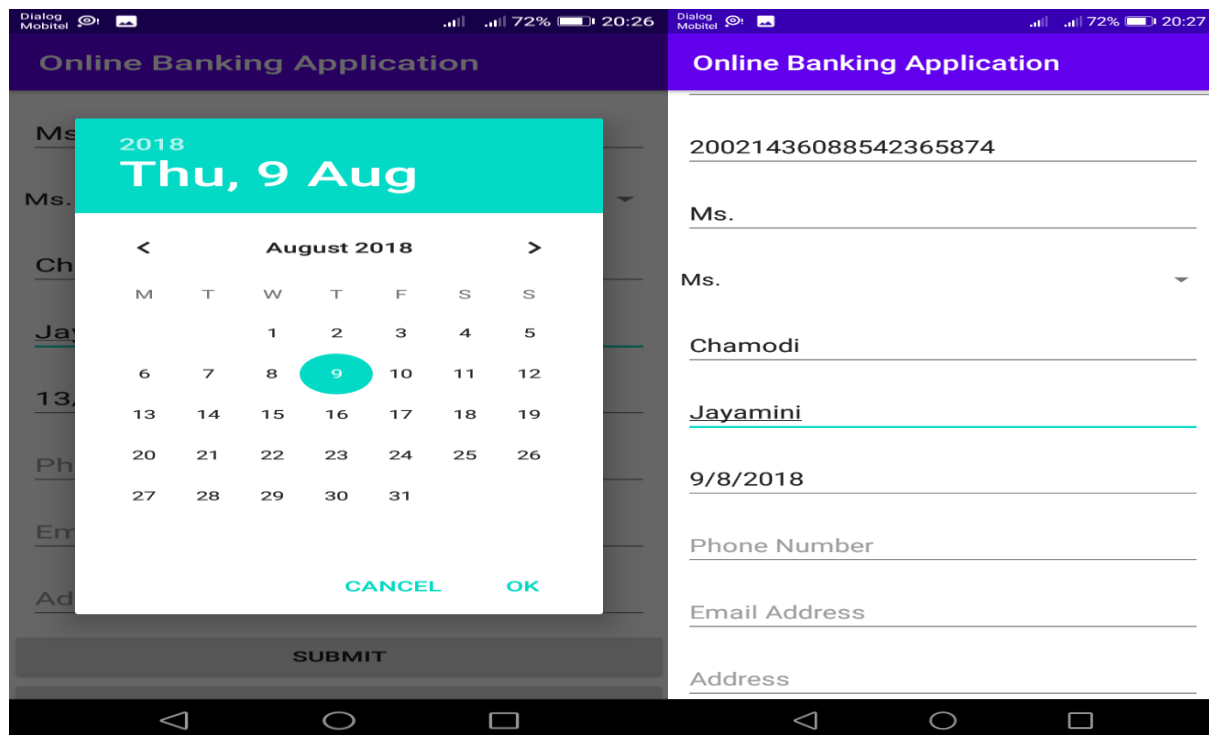- **Date of Birth (Use Data picker)**
  In activity_main.xml,

```xml
<EditText
    android:id="@+id/dob"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:ems="10"
    android:clickable="true"
    android:focusable="false"
    android:hint="Date of Birth"
    android:inputType="date"/>
```

In Main Activity we did not get the future data value set for birthday. It is defined in this code statement.

```java
e8.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final Calendar cldr = Calendar.getInstance();
        int day = cldr.get(Calendar.DAY_OF_MONTH);
        int month = cldr.get(Calendar.MONTH);
        int year = cldr.get(Calendar.YEAR);

        // date picker dialog
        picker = new DatePickerDialog(MainActivity.this,
                new DatePickerDialog.OnDateSetListener() {
                    @Override
                    public void onDateSet(DatePicker view, int year, int monthOfYear, int dayOfMonth) {
                        e8.setText(dayOfMonth + "/" + (monthOfYear + 1) + "/" + year);
                    }
                }, year, month, day);
        picker.show();
        picker.getDatePicker().setMaxDate(System.currentTimeMillis());
    }
});
```

Screen Output of Data picker,

The future values are not valid here,
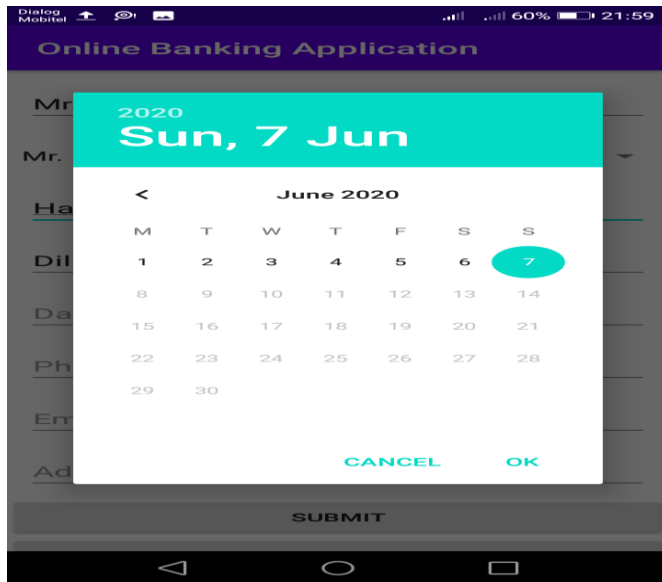
- **Phone number (Use correct android configuration param)**
  In activity_main.xml,

```xml
<EditText
    android:id="@+id/phonenumber"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="Phone Number"
    android:inputType="phone"
    android:digits="0123456789"
    android:maxLength="10"

    />
```

In Main Activity,

```java
String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.+[a-z]+";
EditText emailId;

String pattern = "^\\s*(?:\\+?(\\d{1,3}))?[-. (]*(\\d{3})[-. )]*(\\d{3})[-. ]*(\\d{4})(?: *x(\\d+))?\\s*$";
Matcher m;
DatePickerDialog picker;

Spinner spinner;
EditText text;
```

- **Email address (Use correct android configuration param)**

```xml
<EditText
    android:id="@+id/emailaddress"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="Email Address"
    android:inputType="textEmailAddress"
    />
```

- **Address(Text 250 character)**

```xml
<EditText
    android:id="@+id/address"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="Address"
    android:inputType="text"
    android:maxLength="250"
    />
```

Validation Method

- **Using single validation method in all fields when the submit button is pressed**

```java
b1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(text.getText().toString().isEmpty()||e6.getText().toString().isEmpty()||e1.getText().toString().isEmpty()||e2.getText().toString().isEmpty()||e3.
            Toast.makeText(getApplicationContext(), text: "Fields are empty", Toast.LENGTH_SHORT).show();
        }else {
            if (e6.getText().toString().trim().matches(emailPattern)&& e5.getText().toString().trim().matches(pattern)) {

                builder.setTitle("Confirm dialog Box !");
                builder.setMessage("You want to confirm the input for preceding to another activity?");
                builder.setCancelable(false);
                builder.setPositiveButton( text: "Yes", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        Toast.makeText(getApplicationContext(), text: "You've choosen to preceding to another activity", Toast.LENGTH_SHORT).show();
                        // startActivity(new Intent(MainActivity.this, MyOtherActivity.class));
                        Intent activity2Intent = new Intent(getApplicationContext(), Activity2.class);
                        startActivity(activity2Intent);
                    }
                });

                builder.setNegativeButton( text: "No", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        Toast.makeText(getApplicationContext(), text: "You've changed your mind ", Toast.LENGTH_SHORT).show();
                    }
                });

                builder.show();
                // Toast.makeText(getApplicationContext(),"valid ",Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(getApplicationContext(), text: "Invalid email address or phone number", Toast.LENGTH_SHORT).show();
            }
        }
    }
});
```

If fields are not completed, "Fields are empty" message pop up.

If input are wrong a message will be popup,
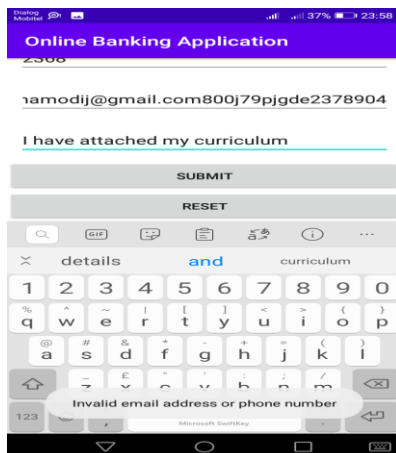
activity_main.xml submit button event,

- **Everything is cleared by single reset button pressed**

```java
//clear all text fields
Button reset = (Button)findViewById(R.id.reset);
reset.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        e1.setText("");
        e2.setText("");
        e3.setText("");
        e4.setText("");
        e5.setText("");
        e6.setText("");
        e7.setText("");
        e8.setText("");
        text.setText("");

    }
});
```

activity_main.xml reset button event,

```xml
<Button
    android:id="@+id/reset"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="Reset"
    android:text="Reset" />
```
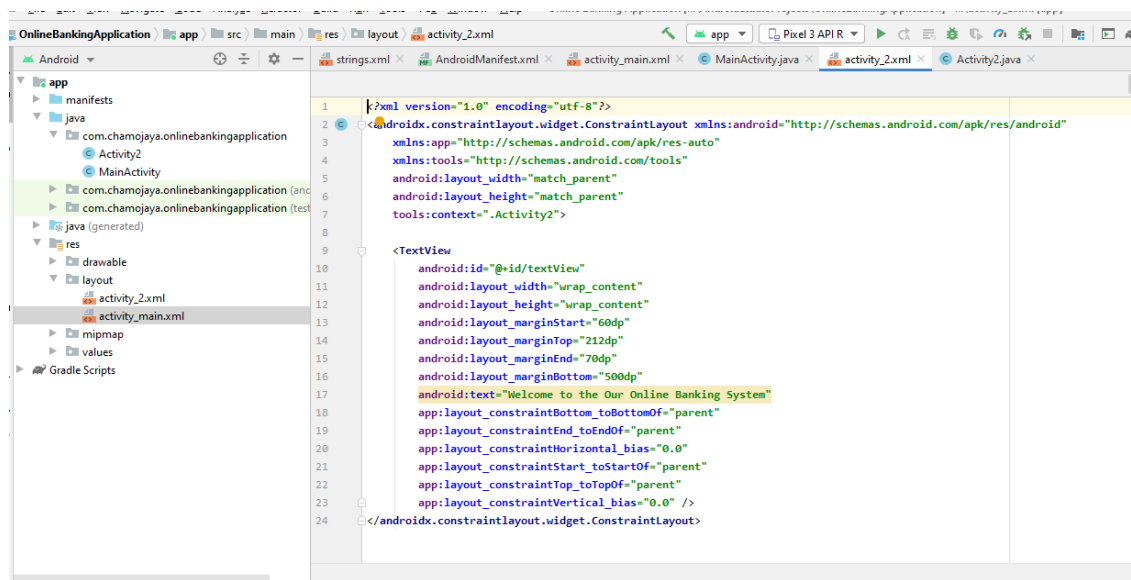
- **If there are no errors a confirmation dialog display when submit button clicked**
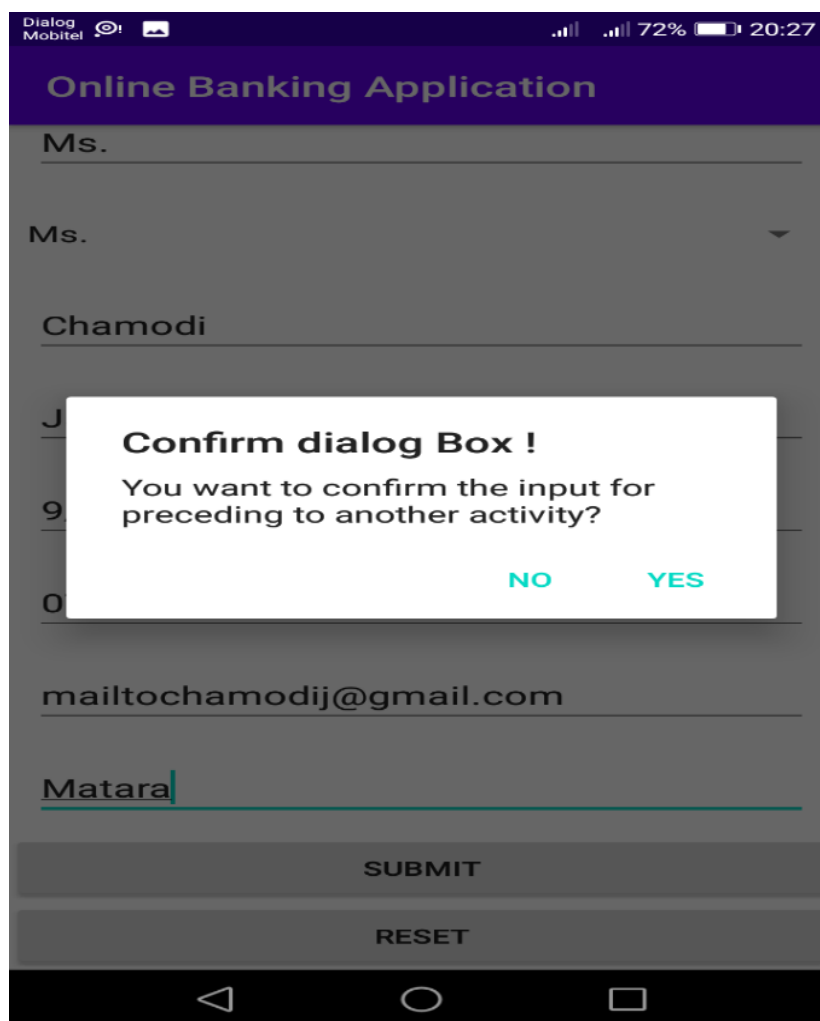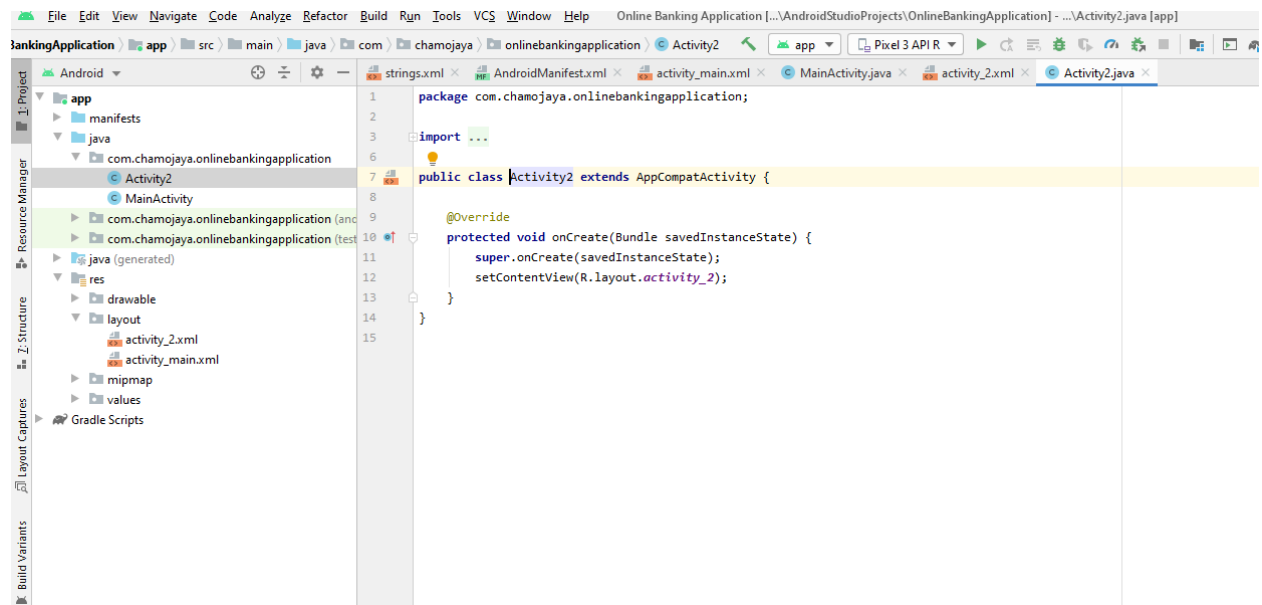
IF yes clicked, a new window came after confirm the inputs with a message "You've choosen to preceding to another activity",
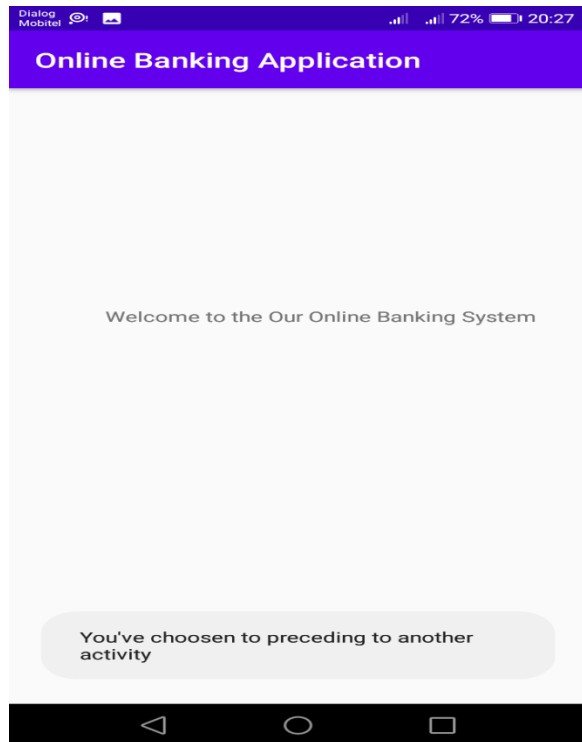
The another activity is proceeding from the main activity,

If No click, a previous window will be refresh with a message "You've changed your mind"

When reset button Clicked,

I used scroll bar in here to go up and down in the screen,

Output

MainActivity.java file

```java
package com.chamojaya.onlinebankingapplication;

import androidx.appcompat.app.AppCompatActivity;

import android.app.Activity;
import android.app.AlertDialog;
import android.app.AlertDialog.Builder;
import android.app.DatePickerDialog;
import android.app.Dialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.telephony.PhoneNumberFormattingTextWatcher;
import android.telephony.PhoneNumberUtils;
import android.text.Editable;
import android.text.TextUtils;
import android.text.TextWatcher;
import android.util.Patterns;
import android.view.Gravity;
```

```java
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.Spinner;
import android.widget.Toast;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.List;
import java.util.Locale;
import java.util.logging.ErrorManager;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

import static android.view.Gravity.*;

public class MainActivity extends AppCompatActivity {

    String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.+[a-z]+";
    EditText emailId;

    String pattern = "^\\s*(?:\\+?(\\d{1,3}))?[-. (]*(\\d{3})[-. )]*(\\d{3})[-.
]*(\\d{4})(?: *x(\\d+))?\\s*$";
    Matcher m;
    DatePickerDialog picker;

    Spinner spinner;
    EditText text;

    List<String> list;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);


        final EditText e1 = (EditText) findViewById(R.id.nationalid);
        final EditText e2 = (EditText) findViewById(R.id.accountnumber);

        final EditText e3 = (EditText) findViewById(R.id.firstname);
        final EditText e4 = (EditText) findViewById(R.id.lastname);
        final EditText e5 = (EditText) findViewById(R.id.phonenumber);
        final EditText e6 = (EditText) findViewById(R.id.emailaddress);
        final EditText e7 = (EditText) findViewById(R.id.address);
        final EditText e8 = (EditText) findViewById(R.id.dob);

        final Button b1 = findViewById(R.id.submit);
```

```java
spinner = (Spinner) findViewById(R.id.spinner);
text = (EditText) findViewById(R.id.salutation);

list = new ArrayList<String>();
list.add("Mr.");
list.add("Ms.");
list.add("Prof.");
list.add("Dr.");
list.add("Rev.");
list.add("Other.");



ArrayAdapter<String> adp = new ArrayAdapter<String>
        (this, android.R.layout.simple_dropdown_item_1line, list);
adp.setDropDownViewResource(android.R.layout.simple_dropdown_item_1line);

spinner.setAdapter(adp);

spinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

    @Override
    public void onItemSelected(AdapterView<?> arg0, View arg1,
                                int arg2, long arg3) {
        // TODO Auto-generated method stub

        switch (arg2) {


            case 0:
                text.setText("Mr.");
                break;
            case 1:
                text.setText("Ms.");
                break;
            case 2:
                text.setText("Prof.");
                break;
            case 3:
                text.setText("Dr.");
                break;
            case 4:
                text.setText("Rev.");
                break;
            default:
                text.setText(" ");
                break;
        }
    }

    @Override
    public void onNothingSelected(AdapterView<?> arg0) {
        // TODO Auto-generated method stub

    }
```

17

```java
        });


        //first element should not be zero
        e2.addTextChangedListener(new TextWatcher(){
            public void onTextChanged(CharSequence s, int start, int before, int
count)
            {
                if (e2.getText().toString().matches("^0") )
                {
                    // Not allowed

                    Toast.makeText(getApplicationContext(),"NOT ALLOWED",
Toast.LENGTH_LONG).show();
                    e2.setText("");
                }
            }
            @Override
            public void afterTextChanged(Editable arg0) { }
            @Override
            public void beforeTextChanged(CharSequence s, int start, int count, int
after) {}
        });


        e8.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                final Calendar cldr = Calendar.getInstance();
                int day = cldr.get(Calendar.DAY_OF_MONTH);
                int month = cldr.get(Calendar.MONTH);
                int year = cldr.get(Calendar.YEAR);

                // date picker dialog
                picker = new DatePickerDialog(MainActivity.this,
                        new DatePickerDialog.OnDateSetListener() {
                            @Override
                            public void onDateSet(DatePicker view, int year, int
monthOfYear, int dayOfMonth) {
                                e8.setText(dayOfMonth + "/" + (monthOfYear + 1) + "/"
+ year);
                            }
                        }, year, month, day);
                picker.show();
                picker.getDatePicker().setMaxDate(System.currentTimeMillis());
            }
        });

        //Validating email address and phone number
        final AlertDialog.Builder builder = new AlertDialog.Builder(this);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

18

```java
if(text.getText().toString().isEmpty()||e6.getText().toString().isEmpty()||e1.getText
().toString().isEmpty()||e2.getText().toString().isEmpty()||e3.getText().toString().i
sEmpty()||e4.getText().toString().isEmpty()||e5.getText().toString().isEmpty()||e7.ge
tText().toString().isEmpty()||e8.getText().toString().isEmpty()){
                    Toast.makeText(getApplicationContext(),"Fields are empty",
Toast.LENGTH_SHORT).show();
                }else {
                    if (e6.getText().toString().trim().matches(emailPattern)&&
e5.getText().toString().trim().matches(pattern)) {

                        builder.setTitle("Confirm dialog Box !");
                        builder.setMessage("You want to confirm the input for
preceding to another activity?");
                        builder.setCancelable(false);
                        builder.setPositiveButton("Yes", new
DialogInterface.OnClickListener() {
                            @Override
                            public void onClick(DialogInterface dialog, int which) {
                                Toast.makeText(getApplicationContext(), "You've
choosen to preceding to another activity", Toast.LENGTH_SHORT).show();
                                // startActivity(new Intent(MainActivity.this,
MyOtherActivity.class));

                                Intent activity2Intent = new
Intent(getApplicationContext(), Activity2.class);
                                startActivity(activity2Intent);
                            }
                        });

                        builder.setNegativeButton("No", new
DialogInterface.OnClickListener() {
                            @Override
                            public void onClick(DialogInterface dialog, int which) {
                                Toast.makeText(getApplicationContext(), "You've
changed your mind ", Toast.LENGTH_SHORT).show();
                            }
                        });

                        builder.show();
                        // Toast.makeText(getApplicationContext(),"valid
",Toast.LENGTH_SHORT).show();
                    } else {
                        Toast.makeText(getApplicationContext(),"Invalid email address
or phone number", Toast.LENGTH_SHORT).show();
                    }
                }
            }
        });
```

```java
        //clear all text fields
        Button reset = (Button)findViewById(R.id.reset);
        reset.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                e1.setText("");
                e2.setText("");
                e3.setText("");
                e4.setText("");
                e5.setText("");
                e6.setText("");
                e7.setText("");
                e8.setText("");
                text.setText("");

            }
        });



    }


}
```