```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">


    <uses-feature
        android:name="android.hardware.camera"
        android:required="false" />

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"
        android:maxSdkVersion="32" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission android:name="com.google.android.c2dm.permission.RECEIVE"
/>
    <uses-permission android:name="com.google.android.c2dm.permission.SEND"/>


    <supports-screens android:resizeable="true"
        android:anyDensity="true"
        android:largeScreens="true"
        android:normalScreens="true"
        android:smallScreens="true"
        android:xlargeScreens="true">
    </supports-screens>


    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.MyApplication.NoActionBar"
        tools:targetApi="31">
        <activity
            android:name=".View.MainActivity"
            android:exported="true"/>

        <activity
            android:name=".Auth_Controller.SignUp"
            android:theme="@style/Theme.MyApplication.NoActionBar" />

        <activity
            android:name=".Auth_Controller.LoginActivity"
            android:exported="true"
            android:theme="@style/Theme.MyApplication.NoActionBar">

            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
```

```xml
        <activity
            android:name=".Auth_Controller.ResetPasswordActivity"
            android:theme="@style/Theme.MyApplication.NoActionBar" />
        <activity
            android:name=".Attempt_Quiz_Section.AttemptTest"
            android:theme="@style/Theme.MyApplication.NoActionBar" />
        <activity
            android:name=".Attempt_Quiz_Section.Tests"
            android:label="@string/title_activity_tests"
            android:theme="@style/Theme.MyApplication.NoActionBar" />

        <activity
            android:name=".Results_section.ResultsAdmin"
            android:label="Results:ADMIN"
            android:theme="@style/Theme.MyApplication.NoActionBar" />
        <activity
            android:name=".Results_section.ResultsAdminDetailed"
            android:theme="@style/Theme.MyApplication.NoActionBar" />


        <activity
            android:name=".Create_Quiz.create_quiz_main"
            android:theme="@style/Theme.MyApplication.NoActionBar">
        </activity>

        <activity
            android:name=".Create_Quiz.Custom_quiz"
            android:theme="@style/Theme.AppCompat.NoActionBar">
        </activity>

        <activity android:name=".Results_section.GetDetailReport"
            android:label="Detailed Report"
            android:theme="@style/Theme.MyApplication.NoActionBar">
        </activity>

    </application>

</manifest>
```

```java
package com.example.myapplication.Attempt_Quiz_Section;

import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Point;
import android.os.Bundle;
import android.os.CountDownTimer;
import androidx.annotation.ColorInt;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
```

```java
import androidx.appcompat.widget.Toolbar;
import androidx.recyclerview.widget.RecyclerView;
import androidx.viewpager2.widget.ViewPager2;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.view.animation.TranslateAnimation;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.BaseAdapter;
import android.widget.Button;
import android.widget.GridView;
import android.widget.LinearLayout;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;

import com.example.myapplication.R;
import com.example.myapplication.Model.Question;
import com.example.myapplication.Model.Test;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

import java.util.ArrayList;
import java.util.concurrent.TimeUnit;

import es.dmoral.toasty.Toasty;

public class AttemptTest extends AppCompatActivity {
    ArrayList<Question> questions;
    String []answers;
    Toolbar toolbar;
    ViewPager2 scrollView;
    LinearLayout indexLayout;
    GridView quesGrid;
    ArrayList<String> list;
    ArrayList<String> arrayList;
    int flag_controller = 1;
    long timer;
    popGridAdapter popGrid;
    Button next, prev;
    TextView textView;
    private DatabaseReference mDatabase;
    private FirebaseAuth auth;
    private String TESTNAME;
    private RadioGroup group;
    private int countPaused = 0;

    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        mDatabase = FirebaseDatabase.getInstance().getReference();
        auth = FirebaseAuth.getInstance();
        setContentView(R.layout.activity_attempt);
        questions = ((Test)
```

```java
getIntent().getExtras().get("Questions")).getQuestions();
        TESTNAME = (String) getIntent().getExtras().get("TESTNAME");
        toolbar = findViewById(R.id.toolbar);

toolbar.setTitleTextColor(getResources().getColor(android.R.color.black));
        answers = new String[questions.size()];
        setSupportActionBar(toolbar);

toolbar.setTitleTextColor(getResources().getColor(android.R.color.black));

        // Изменено на ViewPager2
        scrollView = findViewById(R.id.discrete);
        scrollView.setOrientation(ViewPager2.ORIENTATION_HORIZONTAL);

        final QuestionAdapter questionAdapter = new QuestionAdapter(questions);
        scrollView.setAdapter(questionAdapter);

        next = findViewById(R.id.next);
        next.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                // Изменено для ViewPager2
                if (scrollView.getCurrentItem() == questions.size() - 1) {
                    showPopUp();
                } else {
                    scrollView.setCurrentItem(scrollView.getCurrentItem() + 1);
                }
            }
        });

        prev = findViewById(R.id.prev);
        prev.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                // Изменено для ViewPager2
                if (scrollView.getCurrentItem() != 0) {
                    scrollView.setCurrentItem(scrollView.getCurrentItem() - 1);
                }
            }
        });

        // Изменено для ViewPager2
        setNextPrevButton(scrollView.getCurrentItem());
        indexLayout = findViewById(R.id.index_layout);
        indexLayout.setAlpha(.5f);
        quesGrid = findViewById(R.id.pop_grid);
        popGrid = new popGridAdapter(AttemptTest.this);
        quesGrid.setAdapter(popGrid);
        quesGrid.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> adapterView, View view, int
i, long l) {
                // Изменено для ViewPager2
                scrollView.setCurrentItem(i);
                slideUp(indexLayout);
            }
        });

        // Добавлен слушатель страниц для ViewPager2
```

```java
        scrollView.registerOnPageChangeCallback(new
ViewPager2.OnPageChangeCallback() {
            @Override
            public void onPageSelected(int position) {
                super.onPageSelected(position);
                setNextPrevButton(position);
            }
        });

        timer = ((Test) getIntent().getExtras().get("Questions")).getTime() * 60
* 1000;
    }

    void showPopUp() {
        AlertDialog.Builder builder = new AlertDialog.Builder(AttemptTest.this);
        builder.setMessage("Do you want to submit?");
        builder.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {
                submit();
                dialogStart();
            }
        });

        builder.setNegativeButton("No", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {
                dialogInterface.cancel();
            }
        });
        builder.show();
    }

    /*submit result to database**/
    void submit() {
        flag_controller = 0;
        int score = 0;
        list = new ArrayList<>();
        arrayList = new ArrayList<>();
        for (int i = 0; i < answers.length; i++) {
            if (answers[i] != null &&
answers[i].equals(questions.get(i).getAnswer())) {
                score++;
            }
            String temp = (answers[i] != null) ? answers[i] + ")" : "null) ";

            list.add("Your choice (" +
                    temp +
                    "Right choice is(" + questions.get(i).getAnswer() + ")");
            arrayList.add(questions.get(i).getQuestion());
        }

        try {
            mDatabase.child("Results").child(((Test)
getIntent().getExtras().get("Questions")).getName())
                    .child(auth.getUid()).setValue(score);
        } catch (Exception e) {
            Log.e("Result Update Failed ", e.getMessage());
        }
```

```java
    }

    void dialogStart() {

        final AlertDialog.Builder builderSingle = new
AlertDialog.Builder(AttemptTest.this);
        builderSingle.setIcon(R.mipmap.ic_launcher_round);
        builderSingle.setTitle(TESTNAME + " Answers");
        final ArrayAdapter<String> arrayAdapter = new ArrayAdapter<>
                (AttemptTest.this, android.R.layout.select_dialog_singlechoice);
        final ArrayAdapter<String> arrayAdapter1 = new ArrayAdapter<>
                (AttemptTest.this, android.R.layout.select_dialog_singlechoice);

        for (String y : arrayList) {
            arrayAdapter1.add(y);
        }
        for (String x : list) {
            arrayAdapter.add(x);
        }

        builderSingle.setCancelable(false);
        builderSingle.setNegativeButton("Done!", new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                finish();
                dialog.dismiss();
            }
        });

        builderSingle.setAdapter(arrayAdapter1, new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                String strName = arrayAdapter.getItem(which);
                AlertDialog.Builder builderInner = new
AlertDialog.Builder(AttemptTest.this);
                builderInner.setMessage(strName);
                builderInner.setCancelable(false);
                builderInner.setTitle("Your Selected Question Answer is");
                builderInner.setPositiveButton("Ok", new
DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        builderSingle.show();
                    }
                });
                builderInner.show();
            }
        });
        builderSingle.show();
    }

    @Override
    protected void onPause() {
        super.onPause();
        if (countPaused <= 2 && countPaused >= 0 && flag_controller == 1) {
            Toasty.info(AttemptTest.this, "Finish or submit your test! Before
time runs out.", Toasty.LENGTH_LONG).show();
```

```java
        }
        countPaused++;
    }

    @Override
    protected void onResume() {
        super.onResume();
        if (countPaused > 2) {
            Toasty.success(AttemptTest.this, "Thank you! Your response has been
submitted.", Toasty.LENGTH_SHORT).show();
            countPaused = -1000;
            submit();
            dialogStart();
        }
    }

    @Override
    protected void onRestart() {
        super.onRestart();
        Toasty.info(AttemptTest.this, "Finish or submit your test! Before time
runs out.", Toasty.LENGTH_LONG).show();
    }

    void setNextPrevButton(int pos) {
        if (pos == 0) {
            prev.setText("");
        } else {
            prev.setText("Previous");
        }
        if (pos == questions.size() - 1) {
            next.setText("Submit");
        } else {
            next.setText("Next");
        }
    }

    @Override
    public void onBackPressed() {
        super.onBackPressed();
        showPopUp();
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {

        super.onCreateOptionsMenu(menu);

        getMenuInflater().inflate(R.menu.attempt_menu, menu);
        final MenuItem counter = menu.findItem(R.id.counter);

        new CountDownTimer(timer, 1000) {
            public void onTick(long millisUntilFinished) {
                long millis = millisUntilFinished;
                long hr = TimeUnit.MILLISECONDS.toHours(millis), mn =
(TimeUnit.MILLISECONDS.toMinutes(millis) -

TimeUnit.HOURS.toMinutes(TimeUnit.MILLISECONDS.toHours(millis))),
                        sc = TimeUnit.MILLISECONDS.toSeconds(millis) -
```

```java
                    TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes(millis));

                    String hms = format(hr) + ":" + format(mn) + ":" + format(sc);
                    counter.setTitle(hms);
                    timer = millis;
                }

                String format(long n) {
                    if (n < 10)
                        return "0" + n;
                    else return "" + n;
                }

                public void onFinish() {
                    submit();
                    dialogStart();
                }
            }.start();

            return true;
        }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.submit) {
            showPopUp();

            return true;
        } else if (id == R.id.info) {
            togglePopUp();
        }
        return super.onOptionsItemSelected(item);
    }

    void togglePopUp() {
        if (indexLayout.getVisibility() == View.GONE) {
            slideDown(indexLayout);
        } else slideUp(indexLayout);
    }

    class QuestionAdapter extends
RecyclerView.Adapter<QuestionAdapter.ViewHolder> {

        private int itemHeight;
        private ArrayList<Question> data;

        QuestionAdapter(ArrayList<Question> data) {
            this.data = data;
        }

        @Override
        public void onAttachedToRecyclerView(RecyclerView recyclerView) {
            super.onAttachedToRecyclerView(recyclerView);
            Activity context = (Activity) recyclerView.getContext();
            Point windowDimensions = new Point();

context.getWindowManager().getDefaultDisplay().getSize(windowDimensions);
            itemHeight = Math.round(windowDimensions.y * 0.6f);
```

```java
        }

        @Override
        public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
            LayoutInflater inflater = LayoutInflater.from(parent.getContext());
            View v = inflater.inflate(R.layout.frag_test, parent, false);
            ViewGroup.LayoutParams params = new ViewGroup.LayoutParams(
                    ViewGroup.LayoutParams.MATCH_PARENT,
                    itemHeight);
            v.setLayoutParams(params);
            return new ViewHolder(v);
        }

        @Override
        public void onBindViewHolder(final ViewHolder holder, final int
position) {
            holder.questionText.setText(data.get(position).getQuestion());
            holder.r1.setText(data.get(position).getOpt_A());
            holder.r2.setText(data.get(position).getOpt_B());
            holder.r3.setText(data.get(position).getOpt_C());
            holder.r4.setText(data.get(position).getOpt_D());
            holder.r5.setText("Clear Selected");

            holder.radioGroup.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
                @Override
                public void onCheckedChanged(RadioGroup radioGroup, int i) {
                    final int selectedId =
holder.radioGroup.getCheckedRadioButtonId();
                    int adapterPosition = holder.getAdapterPosition();
                    if (adapterPosition != RecyclerView.NO_POSITION) {
                        if (i == R.id.radioButton) {
                            answers[adapterPosition] = "A";
                        } else if (i == R.id.radioButton2) {
                            answers[adapterPosition] = "B";
                        } else if (i == R.id.radioButton3) {
                            answers[adapterPosition] = "C";
                        } else if (i == R.id.radioButton4) {
                            answers[adapterPosition] = "D";
                        } else if (i == R.id.radioButton5) {
                            holder.radioGroup.clearCheck();
                            answers[adapterPosition] = null;
                        }
                        popGrid.notifyDataSetChanged();
                    }
                }
            });

            if (answers[position] == null) {
                holder.radioGroup.clearCheck();
            } else if (answers[position].equals("A")) {
                holder.radioGroup.check(R.id.radioButton);
            } else if (answers[position].equals("B")) {
                holder.radioGroup.check(R.id.radioButton2);
            } else if (answers[position].equals("C")) {
                holder.radioGroup.check(R.id.radioButton3);
            } else if (answers[position].equals("D")) {
                holder.radioGroup.check(R.id.radioButton4);
            }
```

```java
        }

        @Override
        public int getItemCount() {
            return data.size();
        }

        class ViewHolder extends RecyclerView.ViewHolder {

            private View overlay;
            private TextView questionText;
            private RadioGroup radioGroup;
            private RadioButton r1, r2, r3, r4, r5;

            ViewHolder(View itemView) {
                super(itemView);
                questionText = itemView.findViewById(R.id.questionTextView);
                radioGroup = itemView.findViewById(R.id.radioGroup);
                r1 = itemView.findViewById(R.id.radioButton);
                r2 = itemView.findViewById(R.id.radioButton2);
                r3 = itemView.findViewById(R.id.radioButton3);
                r4 = itemView.findViewById(R.id.radioButton4);
                r5 = itemView.findViewById(R.id.radioButton5);
            }

            public void setOverlayColor(@ColorInt int color) {
                overlay.setBackgroundColor(color);
            }

            public void unCheck() {

                int selectedId = radioGroup.getCheckedRadioButtonId();
                if (radioGroup.getCheckedRadioButtonId() == R.id.radioButton) {
                    r1.setChecked(true);
                } else if (radioGroup.getCheckedRadioButtonId() ==
R.id.radioButton2) {
                    r2.setChecked(true);
                } else if (radioGroup.getCheckedRadioButtonId() ==
R.id.radioButton3) {
                    r3.setChecked(true);
                } else if (radioGroup.getCheckedRadioButtonId() ==
R.id.radioButton4) {
                    r4.setChecked(true);
                } else if (radioGroup.getCheckedRadioButtonId() ==
R.id.radioButton5) {
                    r5.setChecked(true);
                }
            }
        }
    }

    class popGridAdapter extends BaseAdapter {
        Context mContext;

        popGridAdapter(Context context) {
            mContext = context;
        }

        @Override
```

```java
        public Object getItem(int i) {
            return null;
        }

        @Override
        public long getItemId(int i) {
            return i;
        }

        @Override
        public int getCount() {
            return questions.size();
        }

        @Override
        public View getView(final int i, View view, ViewGroup viewGroup) {
            View convertView;
            if (view == null) {
                convertView = new Button(mContext);
            } else convertView = view;
            if (answers[i] == null)

(convertView).setBackgroundColor(getResources().getColor(android.R.color.darker_
gray));
            else

(convertView).setBackgroundColor(getResources().getColor(android.R.color.holo_gr
een_dark));

            ((Button) convertView).setText("" + (i + 1));

            (convertView).setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    scrollView.setCurrentItem(i);
                }
            });
            return convertView;
        }
    }

    public void slideUp(View view) {
        TranslateAnimation animate = new TranslateAnimation(
                0,                      // fromXDelta
                0,                      // toXDelta
                0,  // fromYDelta
                -view.getHeight());                 // toYDelta
        animate.setDuration(500);
        view.startAnimation(animate);
        view.setVisibility(View.GONE);
    }

    // slide the view from its current position to below itself
    public void slideDown(View view) {
        view.setVisibility(View.VISIBLE);
        TranslateAnimation animate = new TranslateAnimation(
                0,                      // fromXDelta
                0,                      // toXDelta
                -view.getHeight(),                  // fromYDelta
```

```java
                0); // toYDelta
        animate.setDuration(500);
        view.startAnimation(animate);
    }

    @Override
    protected void onDestroy() {
        submit();
        super.onDestroy();
    }
}
```

```java
package com.example.myapplication.Attempt_Quiz_Section;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.content.ContextCompat;

import android.util.Log;
import android.view.LayoutInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.Filterable;
import android.widget.ImageView;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import com.example.myapplication.R;
import com.example.myapplication.Model.Question;
import com.example.myapplication.Model.Test;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.Objects;

public class Tests extends AppCompatActivity {
    private FirebaseDatabase database;
```

```java
    private DatabaseReference myRef;
    private ProgressBar progressBar;
    private ListView listView;
    private TestAdapter testAdapter;
    private int lastPos = -1;

    ArrayList<Test> tests = new ArrayList<>();

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

        // Update layout file to use ProgressBar instead of
AVLoadingIndicatorView
        Toolbar toolbar = findViewById(R.id.toolbar);

toolbar.setTitleTextColor(getResources().getColor(android.R.color.black));
        setSupportActionBar(toolbar);

        progressBar = findViewById(R.id.loader1);
        progressBar.setVisibility(View.VISIBLE);

        setSupportActionBar(toolbar);
        Objects.requireNonNull(getSupportActionBar())
                .setDisplayHomeAsUpEnabled(true);

        database = FirebaseDatabase.getInstance();
        myRef = database.getReference();

        listView = findViewById(R.id.test_listview);
        testAdapter = new TestAdapter(Tests.this, tests);
        listView.setAdapter(testAdapter);

        getQues();
    }

    @Override
    protected void onRestart() {
        super.onRestart();
        Toast.makeText(Tests.this, "Finish or submit your test! Before time runs
out.", Toast.LENGTH_LONG).show();
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == android.R.id.home) {
            finish();
        }
        return super.onOptionsItemSelected(item);
    }

    public void getQues() {
        myRef.child("tests").addListenerForSingleValueEvent(new
ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                tests.clear();
```

```java
                for (DataSnapshot snapshot : dataSnapshot.getChildren()) {
                    Test t = new Test();
                    t.setName(snapshot.getKey());

t.setTime(Long.parseLong(snapshot.child("Time").getValue().toString()));

                    ArrayList<Question> ques = new ArrayList<>();
                    for (DataSnapshot qSnap :
snapshot.child("Questions").getChildren()) {
                        ques.add(qSnap.getValue(Question.class));
                    }
                    t.setQuestions(ques);
                    tests.add(t);
                }

                testAdapter.dataList = tests;
                testAdapter.notifyDataSetChanged();
                progressBar.setVisibility(View.GONE);

                Log.e("The read success: ", "su" + tests.size());
            }

            @Override
            public void onCancelled(@NonNull DatabaseError databaseError) {
                progressBar.setVisibility(View.GONE);
                Log.e("The read failed: ", databaseError.getMessage());
            }
        });
    }

    class TestAdapter extends ArrayAdapter<Test> implements Filterable {
        private Context mContext;
        ArrayList<Test> dataList;

        public TestAdapter(Context context, ArrayList<Test> list) {
            super(context, 0, list);
            mContext = context;
            dataList = list;
        }

        @SuppressLint("SetTextI18n")
        @NonNull
        @Override
        public View getView(final int position, @Nullable View convertView,
@NonNull ViewGroup parent) {
            View listItem = convertView;
            if (listItem == null)
                listItem =
LayoutInflater.from(mContext).inflate(R.layout.test_item, parent, false);

            ((ImageView) listItem.findViewById(R.id.item_imageView))
                    .setImageDrawable(ContextCompat.getDrawable(mContext,
R.drawable.ic_appicon));

            ((TextView) listItem.findViewById(R.id.item_textView))
                    .setText(dataList.get(position).getName() + " : " +
dataList.get(position).getTime() + "Min");

            ((Button)
```

```java
listItem.findViewById(R.id.item_button)).setText("Attempt");

            listItem.findViewById(R.id.item_button).setOnClickListener(new
View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    Intent intent = new Intent(mContext, AttemptTest.class);
                    intent.putExtra("Questions", dataList.get(position));
                    intent.putExtra("TESTNAME",
dataList.get(position).getName());
                    startActivity(intent);
                }
            });

            Animation animation = AnimationUtils.loadAnimation(getContext(),
                    (position > lastPos) ? R.anim.up_from_bottom :
R.anim.down_from_top);
            listItem.startAnimation(animation);
            lastPos = position;

            return listItem;
        }
    }
}
```

```java
package com.example.myapplication.Auth_Controller;

import android.content.Intent;
import android.os.Bundle;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.Toast;
import com.example.myapplication.View.MainActivity;
import com.example.myapplication.R;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;

public class LoginActivity extends AppCompatActivity {
    private EditText inputEmail, inputPassword;
    private FirebaseAuth auth;
    private ProgressBar progressBar;
    private Button btnSignup, btnLogin, btnReset;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```java
        auth = FirebaseAuth.getInstance();
        if (auth.getCurrentUser() != null) {
            Intent intent = new Intent(LoginActivity.this, MainActivity.class);
            startActivity(intent);
            overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
            finish();
        }
        setContentView(R.layout.activity_login);

        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        inputEmail = findViewById(R.id.email);
        inputPassword = findViewById(R.id.password);
        progressBar = findViewById(R.id.loader1);
        btnSignup = findViewById(R.id.btn_signup);
        btnLogin = findViewById(R.id.btn_login);
        btnReset = findViewById(R.id.btn_reset_password);

        auth = FirebaseAuth.getInstance();

        btnSignup.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(LoginActivity.this, SignUp.class));
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
            }
        });

        btnReset.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(LoginActivity.this,
ResetPasswordActivity.class));
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
            }
        });

        btnLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String email = inputEmail.getText().toString();
                final String password = inputPassword.getText().toString();

                if (TextUtils.isEmpty(email)) {
                    Toast.makeText(getApplicationContext(), "Enter email
address!", Toast.LENGTH_SHORT).show();
                    return;
                }

                if (TextUtils.isEmpty(password)) {
                    Toast.makeText(getApplicationContext(), "Enter password!",
Toast.LENGTH_SHORT).show();
                    return;
                }

                progressBar.setVisibility(View.VISIBLE);
```

```java
                auth.signInWithEmailAndPassword(email, password)
                    .addOnCompleteListener(LoginActivity.this, new
OnCompleteListener<AuthResult>() {
                        @Override
                        public void onComplete(@NonNull Task<AuthResult>
task) {
                            progressBar.setVisibility(View.GONE);
                            if (!task.isSuccessful()) {
                                if (password.length() < 6) {
inputPassword.setError(getString(R.string.minimum_password));
                                } else {
                                    Toast.makeText(LoginActivity.this,
getString(R.string.auth_failed), Toast.LENGTH_LONG).show();
                                }
                            } else {
                                startActivity(new Intent(LoginActivity.this,
MainActivity.class));

overridePendingTransition(android.R.anim.fade_in, android.R.anim.fade_out);
                                finish();
                            }
                        }
                    });
            }
        });
    }
}
```

```java
package com.example.myapplication.Auth_Controller;

import android.os.Bundle;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.Toast;
import com.example.myapplication.R;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;

public class ResetPasswordActivity extends AppCompatActivity {
    private EditText inputEmail;
    private Button btnReset, btnBack;
    private FirebaseAuth auth;
    private ProgressBar progressBar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

```java
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_reset_password);
        inputEmail = findViewById(R.id.email);
        btnReset = findViewById(R.id.btn_reset_password);
        btnBack = findViewById(R.id.btn_back);
        progressBar = findViewById(R.id.progressBar);
        auth = FirebaseAuth.getInstance();

        btnBack.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                finish();
            }
        });

        btnReset.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String email = inputEmail.getText().toString().trim();
                if (TextUtils.isEmpty(email)) {
                    Toast.makeText(getApplicationContext(), "Enter your
registered email id", Toast.LENGTH_SHORT).show();
                    return;
                }

                progressBar.setVisibility(View.VISIBLE);

                auth.sendPasswordResetEmail(email)
                        .addOnCompleteListener(new OnCompleteListener<Void>() {
                            @Override
                            public void onComplete(@NonNull Task<Void> task) {
                                progressBar.setVisibility(View.GONE);
                                if (task.isSuccessful()) {
                                    Toast.makeText(ResetPasswordActivity.this,
"We have sent you instructions to reset your password!",
Toast.LENGTH_SHORT).show();
                                } else {
                                    Toast.makeText(ResetPasswordActivity.this,
"Failed to send reset email!", Toast.LENGTH_SHORT).show();
                                }
                            }
                        });
            }
        });
    }
}
```

```java
package com.example.myapplication.Auth_Controller;

import android.content.Intent;
import android.os.Bundle;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.text.TextUtils;
```

```java
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.Toast;
import com.example.myapplication.R;
import com.example.myapplication.View.MainActivity;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;

public class SignUp extends AppCompatActivity {
    private EditText inputEmail, inputPassword;
    private Button btnSignIn, btnSignUp, btnResetPassword;
    private ProgressBar progressBar;
    private FirebaseAuth auth;

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

        auth = FirebaseAuth.getInstance();

        btnSignIn = findViewById(R.id.sign_in_button);
        btnSignUp = findViewById(R.id.sign_up_button);
        inputEmail = findViewById(R.id.email);
        inputPassword = findViewById(R.id.password);
        progressBar = findViewById(R.id.loader1);
        btnResetPassword = findViewById(R.id.btn_reset_password);

        btnResetPassword.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(SignUp.this,
ResetPasswordActivity.class));
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
            }
        });

        btnSignIn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
                finish();
            }
        });

        btnSignUp.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String email = inputEmail.getText().toString().trim();
                String password = inputPassword.getText().toString().trim();

                if (TextUtils.isEmpty(email)) {
                    inputEmail.setError("Enter valid Email");
```

```java
                    return;
                }

                if (TextUtils.isEmpty(password)) {
                    inputPassword.setError("Password should be at least 6
characters!");
                    return;
                }

                if (password.length() < 6) {
                    Toast.makeText(getApplicationContext(), "Password too short,
enter minimum 6 characters!", Toast.LENGTH_SHORT).show();
                    return;
                }

                progressBar.setVisibility(View.VISIBLE);

                auth.createUserWithEmailAndPassword(email, password)
                        .addOnCompleteListener(SignUp.this, new
OnCompleteListener<AuthResult>() {
                            @Override
                            public void onComplete(@NonNull Task<AuthResult>
task) {
                                progressBar.setVisibility(View.GONE);
                                if (!task.isSuccessful()) {
                                    Toast.makeText(SignUp.this, "Authentication
failed." + task.getException(), Toast.LENGTH_SHORT).show();
                                } else {
                                    Toast.makeText(SignUp.this, "Registration
successful!", Toast.LENGTH_SHORT).show();
                                    startActivity(new Intent(SignUp.this,
MainActivity.class));

overridePendingTransition(android.R.anim.fade_in, android.R.anim.fade_out);
                                    finish();
                                }
                            }
                        });
            }
        });
    }

    @Override
    protected void onResume() {
        super.onResume();
        progressBar.setVisibility(View.GONE);
    }
}
```

```java
package com.example.myapplication.Create_Quiz;

import android.content.Intent;
```

```java
import android.os.Bundle;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import com.example.myapplication.R;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import java.util.Objects;

public class create_quiz_main extends AppCompatActivity {
    private FloatingActionButton fab1;
    private Toolbar toolbar;

    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.content_main_1);
        toolbar = findViewById(R.id.toolbartst);

toolbar.setTitleTextColor(getResources().getColor(android.R.color.black));
        setSupportActionBar(toolbar);

Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnabled(true);
        fab1 = findViewById(R.id.fab);
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId() == android.R.id.home) {
            finish();
        }
        return super.onOptionsItemSelected(item);
    }
}
```

```java
package com.example.myapplication.Create_Quiz;

import android.content.DialogInterface;
import android.os.Bundle;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;

import android.text.TextUtils;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.TextView;
import android.widget.Toast;

import com.example.myapplication.R;
```

```java
import com.example.myapplication.Model.Question;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.gson.Gson;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.util.ArrayList;
import java.util.Map;

import es.dmoral.toasty.Toasty;

public class Custom_quiz extends AppCompatActivity {

    EditText question;
    EditText aText;
    EditText bText;
    EditText cText;
    EditText dText;
    RadioButton aRadio;
    RadioButton bRadio;
    RadioButton cRadio;
    RadioButton dRadio;

    int currentQuestion = 1;
    int previousQuestion = 1;
    TextView questionNumber;

    ArrayList<Question> ques;
    JSONArray jsonArray;
    String selectedOption = "";

    Button save_button;
    AlertDialog alertDialog;
    private View dialogvView;
    String fileName = "file";
    private FirebaseDatabase database;
    private FirebaseAuth auth;
    private DatabaseReference myRef;
    CardView fab,f2,fl;

    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        jsonArray = new JSONArray();
        setContentView(R.layout.activity_create_quiz);
        question = findViewById(R.id.questionView);
        question =  findViewById(R.id.questionView);
        aText =  findViewById(R.id.aText);
        bText =  findViewById(R.id.bText);
        cText =  findViewById(R.id.cText);
        dText =  findViewById(R.id.dText);
        questionNumber =  findViewById(R.id.questionNumber);
        aRadio =  findViewById(R.id.aRadio);

        bRadio =  findViewById(R.id.bRadio);
```

```java
        cRadio =  findViewById(R.id.cRadio);
        dRadio =  findViewById(R.id.dRadio);
        auth = FirebaseAuth.getInstance();
        database= FirebaseDatabase.getInstance();
        myRef=database.getReference();
        selectedOption = "";
        currentQuestion = 1;
        setListeners();

        ques = new ArrayList<>();

        alertDialog = new AlertDialog.Builder(this).create();
        LayoutInflater inflater = this.getLayoutInflater();
        dialogvView = inflater.inflate(R.layout.dialog_custom,null);



        fab = findViewById(R.id.nextfab);
        f1 = findViewById(R.id.fab2);//save button
        f2 = findViewById(R.id.pre_card);

        f2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                if(previousQuestion>1) {
                    previousQuestion--;
                    setAllData(previousQuestion);
                }
                if(previousQuestion==1)
                    f2.setVisibility(View.INVISIBLE);
                    //Question question1 = new Question();
                Toast.makeText(Custom_quiz.this,
String.valueOf(previousQuestion), Toast.LENGTH_SHORT).show();
            }
        });

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

                if(previousQuestion!=currentQuestion) {
                    previousQuestion++;
                    if(previousQuestion!=currentQuestion)
                    setAllData(previousQuestion);
                    else {
                        clearAllData();
                        questionNumber.setText(String.valueOf(currentQuestion));
                    }
                    if(previousQuestion>1)
                        f2.setVisibility(View.VISIBLE);
                }
                boolean cont = getEnteredQuestionsValue();
                if (cont)
                {
                    previousQuestion++;
                    currentQuestion++;
                    Toast.makeText(Custom_quiz.this, "QUESTION " +
currentQuestion, Toast.LENGTH_SHORT).show();
```

```java
                    questionNumber.setText(String.valueOf(currentQuestion));
                    clearAllData();
                    f2.setVisibility(View.VISIBLE);
                }
            }
        });

        f1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(jsonArray.length()!=0)
                {
                    final JSONObject tempObject = new JSONObject();
                    // get dialog_custom.xml view
                    LayoutInflater li = LayoutInflater.from(Custom_quiz.this);
                    View promptsView = li.inflate(R.layout.dialog_custom, null);
                    AlertDialog.Builder alertDialogBuilder = new
AlertDialog.Builder(
                            Custom_quiz.this);

                    // set dialog_custom.xml to alertdialog builder
                    alertDialogBuilder.setView(promptsView);
                    final EditText userInput =   promptsView
                            .findViewById(R.id.editTextDialogUserInput);
                    final EditText userTime =
promptsView.findViewById(R.id.editTextDialogUserInput1);

                    // set dialog message
                    alertDialogBuilder
                            .setCancelable(false)
                            .setPositiveButton("OK",
                                    new DialogInterface.OnClickListener() {
                                        public void onClick(DialogInterface
dialog,int id) {

                                            final String str =
userTime.getText().toString();
                                            String temp2 = str;
                                    ////
Log.d("TIMEON",userTime.getText().toString().trim());
                                            try {

tempObject.put("Questions",jsonArray);
                                                final String TIME =
userTime.getText().toString().trim();

tempObject.put("Time",Integer.parseInt(temp2));
                                            } catch (JSONException e) {
                                                e.printStackTrace();
                                            }
                                            final String jsonStr =
tempObject.toString();
                                            // get user input and set it to
result
                                            // edit text
                                            if (str != null ) {

Toast.makeText(getApplicationContext()
                                                    , str,
```

```java
Toast.LENGTH_LONG).show();
//                                                result.setText(userInput.getText());
                                                    Map<String, Object> result = new
Gson().fromJson(jsonStr, Map.class);

                                                    fileName =
userInput.getText().toString().trim();

                                                    if
(!TextUtils.isEmpty(fileName))

myRef.child("tests").child(fileName).setValue(result);
                                                    }
                                                }
                                            })
                                    .setNegativeButton("Cancel",
                                            new DialogInterface.OnClickListener() {
                                                public void onClick(DialogInterface
dialog,int id) {

                                                    dialog.cancel();
                                                }
                                            });
                    // create alert dialog
                    AlertDialog alertDialog = alertDialogBuilder.create();
                    // show it
                    alertDialog.show();
                }
                else
                {
                    Toasty.error(getApplicationContext(),
                            "Incomplete Question format",
Toasty.LENGTH_SHORT).show();
                }
            }
        });

    }

    @Override
    public void onBackPressed() {
        super.onBackPressed();
        final AlertDialog.Builder builder = new
AlertDialog.Builder(Custom_quiz.this);
        builder.setMessage("Exit without saving?");
        builder.setPositiveButton("Ok", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                finish();
            }
        }).setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });
        builder.show();
    }

    public void updateData(int position) {
        Question question1 = new Question();
        question1 = ques.get(position-1);
```

```java
    }

    public void setAllData(int position) {
        clearAllData();
        Question question1 = new Question();
        question1 = ques.get(position-1);
        questionNumber.setText(String.valueOf(question1.getId()));
        question.setText(question1.getQuestion());
        aText.setText(question1.getOpt_A());
        bText.setText(question1.getOpt_B());
        cText.setText(question1.getOpt_C());
        dText.setText(question1.getOpt_D());
        switch (question1.getAnswer()){
            case "A":
                aRadio.setChecked(true);
                break;
            case "B":
                bRadio.setChecked(true);
                break;
            case "C":
                cRadio.setChecked(true);
                break;
            case "D":
                dRadio.setChecked(true);
                break;
        }
    }

    private void clearAllData() {

        aRadio.setChecked(false);
        bRadio.setChecked(false);
        cRadio.setChecked(false);
        dRadio.setChecked(false);
        aText.setText(null);
        bText.setText(null);
        cText.setText(null);
        dText.setText(null);
        question.setText(null);
        selectedOption = "";
    }

    private boolean getEnteredQuestionsValue() {

        boolean cont = false;
        if (TextUtils.isEmpty(question.getText().toString().trim())) {
            question.setError("Please fill in a question");
        }
        else if (TextUtils.isEmpty(aText.getText().toString().trim())) {
            aText.setError("Please fill in option A");
        }
        else if (TextUtils.isEmpty(bText.getText().toString().trim())) {
            bText.setError("Please fill in option B");
        }
        else if (TextUtils.isEmpty(cText.getText().toString().trim())) {
            cText.setError("Please fill in option C");
        }
        else if (TextUtils.isEmpty(dText.getText().toString().trim())) {
            dText.setError("Please fill in option D");
```

```java
            }
        else if (selectedOption.equals("")) {
            Toast.makeText(this, "Please select the correct answer",
Toast.LENGTH_SHORT).show();
        }
        else {
            Question quest = new Question();
            quest.setId(currentQuestion);
            quest.setQuestion(question.getText().toString());
            quest.setOpt_A(aText.getText().toString());
            quest.setOpt_B(bText.getText().toString());
            quest.setOpt_C(cText.getText().toString());
            quest.setOpt_D(dText.getText().toString());
            quest.setAnswer(selectedOption);
            ques.add(quest);
            cont = true;

            JSONObject jsonObject = new JSONObject();
            try {
                jsonObject.put("answer",selectedOption);
                jsonObject.put("opt_A",aText.getText().toString().trim());
                jsonObject.put("opt_B",bText.getText().toString().trim());
                jsonObject.put("opt_C",cText.getText().toString().trim());
                jsonObject.put("opt_D",dText.getText().toString().trim());
                jsonObject.put("question",question.getText().toString().trim());

            } catch (JSONException e) {
                e.printStackTrace();
            }

            jsonArray.put(jsonObject);
        }
        return cont;
    }

    private void setListeners() {
        aRadio.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                selectedOption = "A";
                bRadio.setChecked(false);
                cRadio.setChecked(false);
                dRadio.setChecked(false);
            }
        });
        bRadio.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                selectedOption = "B";
                aRadio.setChecked(false);
                cRadio.setChecked(false);
                dRadio.setChecked(false);
            }
        });
        cRadio.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                selectedOption = "C";
                bRadio.setChecked(false);
```

```java
                aRadio.setChecked(false);
                dRadio.setChecked(false);
            }
        });
        dRadio.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                selectedOption = "D";
                bRadio.setChecked(false);
                cRadio.setChecked(false);
                aRadio.setChecked(false);
            }
        });


    }
}
```

```java
package com.example.myapplication.Misc_Section;

import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.drawable.Drawable;
import android.widget.ImageView;

import androidx.core.graphics.drawable.RoundedBitmapDrawable;
import androidx.core.graphics.drawable.RoundedBitmapDrawableFactory;

import com.bumptech.glide.Glide;
import com.bumptech.glide.load.engine.DiskCacheStrategy;
import com.bumptech.glide.request.RequestListener;
import com.bumptech.glide.request.RequestOptions;
import com.bumptech.glide.request.target.BitmapImageViewTarget;


public class ImageUtils {

    // Prevent instantiation
    private ImageUtils() {

    }

    /**
     * Crops image into a circle that fits within the ImageView.
     */
    public static void displayRoundImageFromUrl(final Context context, final
String url, final ImageView imageView) {
        RequestOptions myOptions = new RequestOptions()
                .centerCrop()
                .dontAnimate();

        Glide.with(context)
                .asBitmap()
                .apply(myOptions)
                .load(url)
```

```java
                .diskCacheStrategy(DiskCacheStrategy.DATA)
                .into(new BitmapImageViewTarget(imageView) {
                    @Override
                    protected void setResource(Bitmap resource) {
                        RoundedBitmapDrawable circularBitmapDrawable =
RoundedBitmapDrawableFactory.create(context.getResources(), resource);
                        circularBitmapDrawable.setCircular(true);
                        imageView.setImageDrawable(circularBitmapDrawable);
                    }
                });
    }

    public static void displayImageFromUrl(final Context context, final String
url,
                                           final ImageView imageView) {
        displayImageFromUrl(context, url, imageView, null, null);
    }

    public static void displayImageFromUrl(final Context context, final String
url,
                                           final ImageView imageView, Drawable
placeholderDrawable) {
        displayImageFromUrl(context, url, imageView, placeholderDrawable, null);
    }

    /**
     * Displays an image from a URL in an ImageView.
     */
    public static void displayImageFromUrl(final Context context, final String
url,
                                           final ImageView imageView, Drawable
placeholderDrawable, RequestListener listener) {
        RequestOptions myOptions = new RequestOptions()
                .dontAnimate()
                .diskCacheStrategy(DiskCacheStrategy.AUTOMATIC)
                .placeholder(placeholderDrawable);

        if (listener != null) {
            Glide.with(context)
                    .load(url)
                    .apply(myOptions)
                    .listener(listener)
                    .into(imageView);
        } else {
            Glide.with(context)
                    .load(url)
                    .apply(myOptions)
                    .listener(listener)
                    .into(imageView);
        }
    }

    public static void displayRoundImageFromUrlWithoutCache(final Context
context, final String url,
                                                            final ImageView
imageView) {
        displayRoundImageFromUrlWithoutCache(context, url, imageView, null);
    }
```

```java
    public static void displayRoundImageFromUrlWithoutCache(final Context
context, final String url,
                                              final ImageView imageView,
RequestListener listener) {
        RequestOptions myOptions = new RequestOptions()
                .centerCrop()
                .dontAnimate()
                .diskCacheStrategy(DiskCacheStrategy.NONE)
                .skipMemoryCache(true);

        if (listener != null) {
            Glide.with(context)
                    .asBitmap()
                    .load(url)
                    .apply(myOptions)
                    .listener(listener)
                    .into(new BitmapImageViewTarget(imageView) {
                        @Override
                        protected void setResource(Bitmap resource) {
                            RoundedBitmapDrawable circularBitmapDrawable =
RoundedBitmapDrawableFactory.create(context.getResources(), resource);
                            circularBitmapDrawable.setCircular(true);
                            imageView.setImageDrawable(circularBitmapDrawable);
                        }
                    });
        } else {
            Glide.with(context)
                    .asBitmap()
                    .load(url)
                    .apply(myOptions)
                    .into(new BitmapImageViewTarget(imageView) {
                        @Override
                        protected void setResource(Bitmap resource) {
                            RoundedBitmapDrawable circularBitmapDrawable =
RoundedBitmapDrawableFactory.create(context.getResources(), resource);
                            circularBitmapDrawable.setCircular(true);
                            imageView.setImageDrawable(circularBitmapDrawable);
                        }
                    });
        }
    }

    /**
     * Displays an image from a URL in an ImageView.
     * If the image is loading or nonexistent, displays the specified
placeholder image instead.
     */
    public static void displayImageFromUrlWithPlaceHolder(final Context context,
final String url,
                                                    final ImageView
imageView,
                                                    int placeholderResId)
{
        RequestOptions myOptions = new RequestOptions()
                .dontAnimate()
                .diskCacheStrategy(DiskCacheStrategy.AUTOMATIC)
                .placeholder(placeholderResId);
```

```java
        Glide.with(context)
                .load(url)
                .apply(myOptions)
                .into(imageView);
    }

    /**
     * Displays an image from a URL in an ImageView.
     */
    public static void displayGifImageFromUrl(Context context, String url,
ImageView imageView, Drawable placeholderDrawable, RequestListener listener) {
        RequestOptions myOptions = new RequestOptions()
                .dontAnimate()
                .diskCacheStrategy(DiskCacheStrategy.AUTOMATIC)
                .placeholder(placeholderDrawable);

        if (listener != null) {
            Glide.with(context)
                    .asGif()
                    .load(url)
                    .apply(myOptions)
                    .listener(listener)
                    .into(imageView);
        } else {
            Glide.with(context)
                    .asGif()
                    .load(url)
                    .apply(myOptions)
                    .into(imageView);
        }
    }

    public static void displayGifImageFromUrl(Context context, String url,
ImageView imageView, String thumbnailUrl) {
        displayGifImageFromUrl(context, url, imageView, thumbnailUrl, null);
    }

    /**
     * Displays an GIF image from a URL in an ImageView.
     */
    public static void displayGifImageFromUrl(Context context, String url,
ImageView imageView, String thumbnailUrl, Drawable placeholderDrawable) {
        RequestOptions myOptions = new RequestOptions()
                .dontAnimate()
                .diskCacheStrategy(DiskCacheStrategy.AUTOMATIC)
                .placeholder(placeholderDrawable);

        if (thumbnailUrl != null) {
            Glide.with(context)
                    .asGif()
                    .load(url)
                    .apply(myOptions)
                    .thumbnail(Glide.with(context).asGif().load(thumbnailUrl))
                    .into(imageView);
        } else {
            Glide.with(context)
                    .asGif()
                    .load(url)
                    .apply(myOptions)
```

```
                        .into(imageView);
            }
    }
}
```

```java
package com.example.myapplication.Model;
import java.io.Serializable;

public class Question implements Serializable{

    private int id;
    private String question;
    private String opt_A;
    private String opt_B;
    private String opt_C;
    private String opt_D;
    private String answer;

    public Question() {

    }
    //constructer for setting values


    public Question(int id, String question, String opt_A, String opt_B, String
opt_C, String opt_D, String answer) {
        this.id = id;
        this.question = question;
        this.opt_A = opt_A;
        this.opt_B = opt_B;
        this.opt_C = opt_C;
        this.opt_D = opt_D;
        this.answer = answer;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getQuestion() {
        return question;
    }

    public void setQuestion(String question) {
        this.question = question;
    }

    public String getOpt_A() {
        return opt_A;
    }
```

```java
    public void setOpt_A(String opt_A) {
        this.opt_A = opt_A;
    }

    public String getOpt_B() {
        return opt_B;
    }

    public void setOpt_B(String opt_B) {
        this.opt_B = opt_B;
    }

    public String getOpt_C() {
        return opt_C;
    }

    public void setOpt_C(String opt_C) {
        this.opt_C = opt_C;
    }

    public String getOpt_D() {
        return opt_D;
    }

    public void setOpt_D(String opt_D) {
        this.opt_D = opt_D;
    }

    public String getAnswer() {
        return answer;
    }

    public void setAnswer(String answer) {
        this.answer = answer;
    }
}
```

```java
package com.example.myapplication.Model;

public class Result {
    private String test, score;

    public String getTest() {
        return test;
    }

    public void setTest(String test) {
        this.test = test;
    }

    public String getScore() {
        return score;
    }
```

```java
    public void setScore(String score) {
        this.score = score;
    }
}
```

```java
package com.example.myapplication.Model;

import java.io.Serializable;
import java.util.ArrayList;

public class Test implements Serializable {
    String Name;
    ArrayList<Question> Questions;
    Long time;

    public Test() {
    }

    public Long getTime() {
        return time;
    }

    public void setTime(Long time) {
        this.time = time;
    }

    public String getName() {
        return Name;
    }

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

    public ArrayList<Question> getQuestions() {
        return Questions;
    }

    public void setQuestions(ArrayList<Question> questions) {
        Questions = questions;
    }
}
```

```java
package com.example.myapplication.Model;

public class User {
    public String name, semester, branch, sect;
```

```java
    public User() {
    }

    public User(String name, String semester, String branch, String sect) {
        this.name = name;
        this.semester = semester;
        this.branch = branch;
        this.sect = sect;
    }

    public String getName() {
        return name;
    }

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

    public String getSemester() {
        return semester;
    }

    public void setSemester(String semester) {
        this.semester = semester;
    }

    public String getBranch() {
        return branch;
    }

    public void setBranch(String branch) {
        this.branch = branch;
    }

    public String getSect() {
        return sect;
    }

    public void setSect(String secti) {
        this.sect = secti;
    }
}
```

```java
package com.example.myapplication.Results_section;

import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.Bundle;
import android.util.DisplayMetrics;
import android.view.MenuItem;
import android.widget.ImageView;
import android.widget.TextView;
```

```java
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;

import com.example.myapplication.R;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;

import java.util.Objects;

import de.hdodenhof.circleimageview.CircleImageView;

public class GetDetailReport extends AppCompatActivity {

    CircleImageView circleImageView;
    TextView textView1;
    TextView textView2;
    TextView textView3;
    TextView textView4;
    TextView textView5;
    private FirebaseDatabase database;
    private FirebaseAuth auth;
    private DatabaseReference myRef;
    private StorageReference firebaseStorage;
    final long ONE_MEGABYTE = 1024 * 1024;


    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.complete_student_details);
        Toolbar toolbar =  findViewById(R.id.toolbartst);
        setSupportActionBar(toolbar);

toolbar.setTitleTextColor(getResources().getColor(android.R.color.black));

Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnabled(true);
        circleImageView = findViewById(R.id.USERIMAGE);
        textView1 = findViewById(R.id.StudentID);
        textView2 = findViewById(R.id.SemID);
        textView3 = findViewById(R.id.Branchid);
        textView4 = findViewById(R.id.SectionID);
        textView5 = findViewById(R.id.this_Section);
        String temp = getIntent().getStringExtra("USERID");

        firebaseStorage =
FirebaseStorage.getInstance().getReference().child(temp);
        firebaseStorage.getBytes(ONE_MEGABYTE)
                .addOnSuccessListener(new OnSuccessListener<byte[]>() {
                    @Override
                    public void onSuccess(byte[] bytes) {
                        Bitmap bm = BitmapFactory.decodeByteArray(bytes, 0,
```

```java
bytes.length);
                            DisplayMetrics dm = new DisplayMetrics();
                            getWindowManager().getDefaultDisplay().getMetrics(dm);

                            circleImageView.setMinimumHeight(100);
                            circleImageView.setMinimumWidth(100);
                            circleImageView.setMaxHeight(100);
                            circleImageView.setMaxWidth(100);

circleImageView.setScaleType(ImageView.ScaleType.CENTER_CROP);
                            circleImageView.setImageBitmap(bm);
                    }
                }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {


            }
        });

        textView1.setText(getIntent().getStringExtra("DetailID"));
        textView2.setText(getIntent().getStringExtra("DetailBranch"));
        textView3.setText(getIntent().getStringExtra("DetailSem"));
        textView4.setText(getIntent().getStringExtra("DetailSec"));
        String Temp = getIntent().getStringExtra("TestNAME") + " results is: " +
                getIntent().getStringExtra("Marks");
        textView5.setText(Temp);
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {

        if(item.getItemId() == android.R.id.home) {
            finish();
        }
        return super.onOptionsItemSelected(item);

    }
}
```

```java
package com.example.myapplication.Results_section;

import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
```

```java
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.TextView;
import com.example.myapplication.R;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import java.util.ArrayList;
import java.util.Objects;

public class ResultsAdmin extends AppCompatActivity {

    private FirebaseDatabase database;
    private FirebaseAuth auth;
    private DatabaseReference myRef;
    private ProgressBar progressBar;
    private ListView listView;
    private TestAdapter testAdapter;
    ArrayList<String> result = new ArrayList<>();
    private int lastPos = -1;
    private boolean isAdmin = false;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        auth = FirebaseAuth.getInstance();
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tests);
        isAdmin = getIntent().getBooleanExtra("ISADMIN", false);
        if (!isAdmin)
            setTitle("Results");
        Toolbar toolbar = findViewById(R.id.toolbar);

toolbar.setTitleTextColor(getResources().getColor(android.R.color.black));
        setSupportActionBar(toolbar);

Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnabled(true);
        progressBar = findViewById(R.id.loader1);
        progressBar.setVisibility(View.VISIBLE);
        database = FirebaseDatabase.getInstance();
        myRef = database.getReference();
        listView = findViewById(R.id.test_listview);
        testAdapter = new TestAdapter(ResultsAdmin.this, result);
        listView.setAdapter(testAdapter);
        getResults();
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId() == android.R.id.home) {
            finish();
        }
```

```java
            return super.onOptionsItemSelected(item);
    }

    public void getResults() {
        if (isAdmin) {
            myRef.child("Results").addListenerForSingleValueEvent(new
ValueEventListener() {
                @Override
                public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                    result.clear();
                    for (DataSnapshot snapshot : dataSnapshot.getChildren()) {
                        result.add(snapshot.getKey());
                    }
                    testAdapter.dataList = result;
                    testAdapter.notifyDataSetChanged();
                    progressBar.setVisibility(View.GONE);
                    Log.e("The read success: ", "su" + result.size());
                }

                @Override
                public void onCancelled(@NonNull DatabaseError databaseError) {
                    progressBar.setVisibility(View.GONE);
                    Log.e("The read failed: ", databaseError.getMessage());
                }
            });
        } else {
            myRef.child("Results").addListenerForSingleValueEvent(new
ValueEventListener() {
                @Override
                public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                    result.clear();
                    for (DataSnapshot snapshot : dataSnapshot.getChildren()) {
                        if
(snapshot.hasChild(Objects.requireNonNull(auth.getUid())))
                            result.add(snapshot.getKey());
                    }
                    testAdapter.dataList = result;
                    testAdapter.notifyDataSetChanged();
                    progressBar.setVisibility(View.GONE);
                    Log.e("The read success: ", "su" + result.size());
                }

                @Override
                public void onCancelled(@NonNull DatabaseError databaseError) {
                    progressBar.setVisibility(View.GONE);
                    Log.e("The read failed: ", databaseError.getMessage());
                }
            });
        }
    }

    class TestAdapter extends ArrayAdapter<String> {

        private Context mContext;
        ArrayList<String> dataList;

        public TestAdapter(Context context, ArrayList<String> list) {
            super(context, 0, list);
            mContext = context;
```

```java
                dataList = list;
            }

        @NonNull
        @Override
        public View getView(final int position, @Nullable View convertView,
@NonNull ViewGroup parent) {
            View listItem = convertView;
            if (listItem == null)
                listItem =
LayoutInflater.from(mContext).inflate(R.layout.test_item, parent, false);
            ((ImageView) listItem.findViewById(R.id.item_imageView))
                    .setImageDrawable(mContext.getDrawable(R.drawable.ranking));
            ((ImageView)
listItem.findViewById(R.id.item_imageView)).setPadding(10, 0, 0, 0);
            ((TextView)
listItem.findViewById(R.id.item_textView)).setText(dataList.get(position));
            ((Button) listItem.findViewById(R.id.item_button)).setText("View");

            Animation animation = AnimationUtils.loadAnimation(getContext(),
                    (position > lastPos) ? R.anim.up_from_bottom :
R.anim.down_from_top);
            listItem.startAnimation(animation);
            lastPos = position;
            ((Button)
listItem.findViewById(R.id.item_button)).setOnClickListener(new
View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    Intent intent = new Intent(ResultsAdmin.this,
ResultsAdminDetailed.class);
                    intent.putExtra("test", dataList.get(position));
                    intent.putExtra("ISAdmin", isAdmin);
                    startActivity(intent);
                }
            });
            return listItem;
        }
    }
}
```

```java
package com.example.myapplication.Results_section;

import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.cardview.widget.CardView;
import androidx.core.content.ContextCompat;

import android.os.Environment;
```

```java
import android.util.Log;
import android.view.LayoutInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import com.example.myapplication.Model.User;
import com.example.myapplication.R;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import org.apache.poi.hssf.usermodel.HSSFWorkbook;
import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;

import java.io.File;
import java.io.FileOutputStream;
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.Objects;

public class ResultsAdminDetailed extends AppCompatActivity {

    private DatabaseReference myRef;
    private ProgressBar progressBar;
    private CardView button;
    private ResultsAdminDetailed.TestAdapter testAdapter;
    ArrayList<TestResults> result = new ArrayList<>();
    private String testName;
    private int lastPos = -1;
    public boolean isAdmin = false;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        FirebaseAuth auth = FirebaseAuth.getInstance();
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tests);
        button = findViewById(R.id.download_card);
        isAdmin = getIntent().getBooleanExtra("ISAdmin", false);
        testName = getIntent().getStringExtra("test");

        if (!isAdmin) {
            setTitle("Result");
        }
```

```java
        if (isAdmin) {
            button.setVisibility(View.VISIBLE);
        }

        Toolbar toolbar = findViewById(R.id.toolbar);

toolbar.setTitleTextColor(getResources().getColor(android.R.color.black));
        setSupportActionBar(toolbar);

Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnabled(true);

        progressBar = findViewById(R.id.loader1);
        progressBar.setVisibility(View.VISIBLE);

        FirebaseDatabase database = FirebaseDatabase.getInstance();
        myRef = database.getReference();

        ListView listView = findViewById(R.id.test_listview);
        testAdapter = new
ResultsAdminDetailed.TestAdapter(ResultsAdminDetailed.this, result);
        listView.setAdapter(testAdapter);
        getSupportActionBar().setTitle(testName);

        getResults();

        button.setOnClickListener(v -> {
            // check if available and not read only
            if (!isExternalStorageAvailable() || isExternalStorageReadOnly()) {
                Log.e("TAG", "Storage not available or read only");
                return;
            }
            saveToExcel(testName.concat(".xls"));
        });
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId() == android.R.id.home) {
            finish();
        }
        return super.onOptionsItemSelected(item);
    }

    public void saveToExcel(String fileName) {
        getResults();
        //New Workbook
        HSSFWorkbook hSSFWorkbook = new HSSFWorkbook();
        Sheet createSheet = hSSFWorkbook.createSheet(this.testName);
        //create row
        Row createRow = createSheet.createRow(0);
        createRow.createCell(0).setCellValue("ID");
        createRow.createCell(1).setCellValue("Semester");
        createRow.createCell(2).setCellValue("Branch");
        createRow.createCell(3).setCellValue("Section");
        createRow.createCell(4).setCellValue("Score");

        int i = 0;
        while (i < this.testAdapter.dataList.size()) {
            int i2 = i + 1;
```

```java
            Row createRow2 = createSheet.createRow(i2);

createRow2.createCell(0).setCellValue(this.testAdapter.dataList.get(i).user.name
);

createRow2.createCell(1).setCellValue(this.testAdapter.dataList.get(i).user.seme
ster);

createRow2.createCell(2).setCellValue(this.testAdapter.dataList.get(i).user.bran
ch);

createRow2.createCell(3).setCellValue(this.testAdapter.dataList.get(i).user.sect
);

createRow2.createCell(4).setCellValue(this.testAdapter.dataList.get(i).score);
            i = i2;
        }

        //save the file under download folder of android
        File file = new
File(Environment.getExternalStoragePublicDirectory(Environment.DIRECTORY_DOWNLOA
DS),fileName);

        try {
            FileOutputStream fileOutputStream = new FileOutputStream(file);
            hSSFWorkbook.write(fileOutputStream);
            StringBuilder sb = new StringBuilder();
            sb.append("Writing file");
            sb.append(fileName);
            Log.w("FileSaver",sb.toString());
            fileOutputStream.close();
            Toast.makeText(ResultsAdminDetailed.this, "File is saved under
Download folder", Toast.LENGTH_SHORT).show();
        } catch (Exception e) {
            Toast.makeText(ResultsAdminDetailed.this, "Can't be saved",
Toast.LENGTH_SHORT).show();
            e.printStackTrace();
        }
    }

    public boolean isExternalStorageReadOnly() {
        return "mounted_ro".equals(Environment.getExternalStorageState());
    }

    public boolean isExternalStorageAvailable() {
        return "mounted".equals(Environment.getExternalStorageState());
    }

    public void getResults() {

myRef.child("Results").child(testName).addListenerForSingleValueEvent(new
ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                result.clear();

                for (DataSnapshot snapshot : dataSnapshot.getChildren()) {
                    TestResults t = new TestResults();
                    t.userID = snapshot.getKey();
```

```java
                    t.score =
Objects.requireNonNull(snapshot.getValue()).toString();
                    result.add(t);
                }

                Collections.sort(result, (o1, o2) ->
o2.score.compareTo(o1.score));

                getDetails();
                Log.e("The read success: ", "su" + result.size());
            }

            @Override
            public void onCancelled(@NonNull DatabaseError databaseError) {
                progressBar.setVisibility(View.GONE);
                Log.e("The read failed: ", databaseError.getMessage());
            }
        });
    }

    private void getDetails() {
        myRef.child("users").addListenerForSingleValueEvent(new
ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                for (int i = 0; i < result.size(); i++) {
                    if (dataSnapshot.child(result.get(i).userID).exists())
                        result.get(i).user =
dataSnapshot.child(result.get(i).userID).getValue(User.class);
                    else {
                        User user1 = new User();
                        user1.setName("Unknown");
                        user1.setBranch("Details not added yet");
                        user1.setSect("Details not added yet");
                        user1.setSemester("Details not added yet");
                        result.get(i).user = user1;
                    }
                }

                testAdapter.dataList = result;
                testAdapter.notifyDataSetChanged();
                progressBar.setVisibility(View.GONE);
            }

            @Override
            public void onCancelled(@NonNull DatabaseError databaseError) {
                progressBar.setVisibility(View.GONE);
            }
        });
    }

    class TestResults {
        public String userID, score;
        public User user;
    }

    class TestAdapter extends ArrayAdapter<TestResults> {
        private Context mContext;
        ArrayList<TestResults> dataList;
```

```java
        public TestAdapter(Context context, ArrayList<TestResults> list) {
            super(context, 0, list);
            mContext = context;
            dataList = list;
        }

        @NonNull
        @Override
        public View getView(final int position, @Nullable View convertView,
@NonNull ViewGroup parent) {
            View listItem = convertView;
            if (listItem == null)
                listItem =
LayoutInflater.from(mContext).inflate(R.layout.test_item, parent, false);

            ((ImageView) listItem.findViewById(R.id.item_imageView))
                    .setImageDrawable(ContextCompat.getDrawable(mContext,
R.drawable.ranking));

            if (dataList.get(position).user != null)
                ((TextView)
listItem.findViewById(R.id.item_textView)).setText(dataList.get(position).user.n
ame);
            else {
                ((TextView)
listItem.findViewById(R.id.item_textView)).setText("Details not added yet");
            }

            Animation animation = AnimationUtils.loadAnimation(getContext(),
(position > lastPos) ? R.anim.up_from_bottom : R.anim.down_from_top);
            (listItem).startAnimation(animation);
            lastPos = position;

            if (isAdmin) {
                ((TextView)
listItem.findViewById(R.id.item_textView)).setOnClickListener(v -> {
                    if (dataList.get(position).user.name != null) {
                        Intent intent = new Intent(ResultsAdminDetailed.this,
GetDetailReport.class);
                        intent.putExtra("USERID",
dataList.get(position).userID);
                        intent.putExtra("DetailID",
dataList.get(position).user.name);
                        intent.putExtra("DetailBranch",
dataList.get(position).user.branch);
                        intent.putExtra("DetailSem",
dataList.get(position).user.semester);
                        intent.putExtra("DetailSec",
dataList.get(position).user.sect);
                        intent.putExtra("TestNAME", testName);
                        intent.putExtra("Marks", dataList.get(position).score);
                        startActivity(intent);
                    }
                });
            }

            ((Button)
listItem.findViewById(R.id.item_button)).setText(dataList.get(position).score);
```

```
            return listItem;
        }
    }
}
```

```java
package com.example.myapplication.View;

import android.app.Dialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.drawable.ColorDrawable;
import android.net.ConnectivityManager;
import android.os.Bundle;
import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.Window;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.RelativeLayout;
import android.widget.TextView;
import android.widget.Toast;
import com.example.myapplication.Attempt_Quiz_Section.Tests;
import com.example.myapplication.Create_Quiz.create_quiz_main;
import com.example.myapplication.Model.User;
import com.example.myapplication.Results_section.ResultsAdmin;
import com.example.myapplication.R;
import com.example.myapplication.Auth_Controller.ResetPasswordActivity;
import com.example.myapplication.Auth_Controller.LoginActivity;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.navigation.NavigationView;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.Objects;
import de.hdodenhof.circleimageview.CircleImageView;

public class MainActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {
```

```java
    private FirebaseDatabase database;
    private FirebaseAuth auth;
    private DatabaseReference myRef;
    public TextView USer_email;
    public ImageButton imageButton;
    private boolean isAdmin = false;
    private TextView userID;
    private CircleImageView imageView;
    public CircleImageView imageView1;
    private FloatingActionButton floatingActionButton;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        imageView = findViewById(R.id.card1);
        floatingActionButton = findViewById(R.id.chatHead);
        auth = FirebaseAuth.getInstance();

        database = FirebaseDatabase.getInstance();
        myRef = database.getReference();

        checkForAdmin();

        DrawerLayout drawer = findViewById(R.id.drawer_layout);
        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
                this, drawer, toolbar, R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
        drawer.addDrawerListener(toggle);
        toggle.syncState();

        NavigationView navigationView = findViewById(R.id.nav_view);
        View header = navigationView.getHeaderView(0);
        imageView1 =
(navigationView.getHeaderView(0)).findViewById(R.id.imageView);
        USer_email = header.findViewById(R.id.text_user_name);
        setTextOnUser();
        navigationView.setNavigationItemSelectedListener(this);
        userID = findViewById(R.id.text_user_card);
        setUserEmail();

        floatingActionButton.setOnClickListener(v -> {
            if (isAdmin) {
                startActivity(new Intent(MainActivity.this,
create_quiz_main.class));
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
            } else {
                Toast.makeText(MainActivity.this, "You are not an admin",
Toast.LENGTH_SHORT).show();
            }
        });
    }

    public void checkForAdmin() {
        myRef.child("admins").addListenerForSingleValueEvent(new
ValueEventListener() {
```

```java
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                if (dataSnapshot.child(Objects.requireNonNull(auth.getUid()))
                        .exists() &&
Objects.requireNonNull(dataSnapshot.child(auth.getUid())
                        .getValue()).toString().equals("true")) {
                    isAdmin = true;
                    Toast.makeText(getApplicationContext(), "Hello Admin",
Toast.LENGTH_LONG).show();
                }
            }

            @Override
            public void onCancelled(@NonNull DatabaseError databaseError) {
            }
        });
    }

    @Override
    public void onBackPressed() {
        DrawerLayout drawer = findViewById(R.id.drawer_layout);
        if (drawer.isDrawerOpen(GravityCompat.START)) {
            drawer.closeDrawer(GravityCompat.START);
        } else {
            super.onBackPressed();
        }
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        return super.onOptionsItemSelected(item);
    }

    @SuppressWarnings("")
    @Override
    public boolean onNavigationItemSelected(MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.nav_test) {
            if (isNetworkAvailable(MainActivity.this)) {
                startActivity(new Intent(MainActivity.this, Tests.class));
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
            } else
                alertNoConnection();
        } else if (id == R.id.nav_result) {
            if (isNetworkAvailable(MainActivity.this)) {
                Intent intent = new Intent(MainActivity.this,
ResultsAdmin.class);
                intent.putExtra("ISADMIN", isAdmin);
                startActivity(intent);
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
```

```java
            } else
                alertNoConnection();
        } else if (id == R.id.create_test) {
            if (isAdmin && isNetworkAvailable(MainActivity.this)) {
                startActivity(new Intent(MainActivity.this,
create_quiz_main.class));
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
            } else if (isNetworkAvailable(MainActivity.this))
                Toast.makeText(getApplicationContext(), "You are not Admin!",
Toast.LENGTH_SHORT).show();
            else
                alertNoConnection();
        } else if (id == R.id.nav_respass) {
            if (isNetworkAvailable(MainActivity.this)) {
                startActivity(new Intent(MainActivity.this,
ResetPasswordActivity.class));
                overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
            } else
                alertNoConnection();
        } else if (id == R.id.nav_signout) {
            auth.signOut();
            startActivity(new Intent(MainActivity.this, LoginActivity.class));
            finish();
            overridePendingTransition(android.R.anim.fade_in,
android.R.anim.fade_out);
        }

        DrawerLayout drawer = findViewById(R.id.drawer_layout);
        drawer.closeDrawer(GravityCompat.START);
        return true;
    }

    public void setUserEmail() {
        DatabaseReference mDatabase;
        FirebaseAuth auth;
        mDatabase = FirebaseDatabase.getInstance().getReference();
        auth = FirebaseAuth.getInstance();
        mDatabase.child("users").child(Objects.requireNonNull(auth.getUid()))
                .addValueEventListener(new ValueEventListener() {
                    @Override
                    public void onDataChange(@NonNull DataSnapshot dataSnapshot)
{
                        if (dataSnapshot.exists()) {
                            User user = dataSnapshot.getValue(User.class);
                            String temp = "Hey " + user.name + " what's up!";
                            userID.setText(temp);
                        }
                    }

                    @Override
                    public void onCancelled(@NonNull DatabaseError
databaseError) {
                    }
                });
    }

    public void setTextOnUser() {
```

```java
        FirebaseUser usero = FirebaseAuth.getInstance().getCurrentUser();
        USer_email.setText(Objects.requireNonNull(usero).getEmail());
    }

    public boolean isNetworkAvailable(Context context) {
        ConnectivityManager connectivityManager = ((ConnectivityManager) context
                .getSystemService(Context.CONNECTIVITY_SERVICE));
        return connectivityManager.getActiveNetworkInfo() != null
                && connectivityManager.getActiveNetworkInfo().isConnected();
    }

    public void alertNoConnection() {
        Dialog builder = new Dialog(this);
        builder.requestWindowFeature(Window.FEATURE_NO_TITLE);
        builder.getWindow().setBackgroundDrawable(
                new ColorDrawable(android.graphics.Color.TRANSPARENT));
        builder.setOnDismissListener(new DialogInterface.OnDismissListener() {
            @Override
            public void onDismiss(DialogInterface dialogInterface) {
            }
        });

        ImageView imageView = new ImageView(this);
        imageView.setImageResource(R.drawable.nowifi);
        builder.addContentView(imageView, new RelativeLayout.LayoutParams(
                400,
                400));
        builder.show();
    }
}
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/Theme.MyApplication.AppBarOverlay">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?android:attr/actionBarSize"
            android:background="?android:attr/colorPrimary"
            app:popupTheme="@style/Theme.MyApplication.PopupOverlay" />
    </com.google.android.material.appbar.AppBarLayout>

    <FrameLayout
```

```xml
        android:layout_width="match_parent"
        android:layout_height="match_parent">

    <androidx.viewpager2.widget.ViewPager2
        android:id="@+id/discrete"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_above="@id/linearLayout"/>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:id="@+id/linearLayout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:padding="5dp"
            android:orientation="horizontal">
            <Button
                style="?android:attr/buttonBarButtonStyle"
                android:layout_marginEnd="5dp"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:textColor="#ffff"
                android:layout_weight="1"
                android:id="@+id/prev"
                android:background="@drawable/circular_coloured_button"/>

            <Button
                style="?android:attr/buttonBarButtonStyle"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:textColor="#ffff"
                android:layout_weight="1"
                android:id="@+id/next"
                android:background="@drawable/circular_coloured_button"/>
        </LinearLayout>
    </RelativeLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="top|center_horizontal"
        android:gravity="center_horizontal"
        android:id="@+id/index_layout"
        android:layout_marginTop="?android:attr/actionBarSize"
        android:orientation="vertical"
        android:visibility="gone">
        <GridView
            android:id="@+id/pop_grid"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:numColumns="4">
        </GridView>
    </LinearLayout>
</FrameLayout>
```

```xml
</RelativeLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">


    <include layout="@layout/content_create_quiz" />

    <RelativeLayout
        android:layout_height="wrap_content"
        android:layout_width="wrap_content"
        android:layout_gravity="end|bottom">
        <androidx.cardview.widget.CardView
            android:layout_height="wrap_content"
            android:layout_width="120dp"
            android:id="@+id/nextfab"
            android:visibility="visible"
            app:cardElevation="10dp"
            app:cardCornerRadius="5dp"
            app:cardBackgroundColor="@color/colorPrimaryDark"
            android:layout_margin="20dp">

            <TextView
                android:layout_height="wrap_content"
                android:layout_width="wrap_content"
                android:layout_margin="10dp"
                android:text="Next"
                android:layout_gravity="center"
                android:textColor="@android:color/black"/>
        </androidx.cardview.widget.CardView>
    </RelativeLayout>

    <RelativeLayout
        android:layout_height="wrap_content"
        android:layout_width="wrap_content"
        android:layout_gravity="start|bottom">
        <androidx.cardview.widget.CardView
            android:layout_height="wrap_content"
            android:layout_width="120dp"
            android:id="@+id/pre_card"
            android:visibility="invisible"
            app:cardElevation="10dp"
            app:cardCornerRadius="5dp"
            app:cardBackgroundColor="@color/colorPrimaryDark"
            android:layout_margin="20dp">
            <TextView
                android:layout_height="wrap_content"
                android:layout_width="wrap_content"
                android:layout_margin="10dp"
                android:layout_gravity="center"
                android:text="Previous"
```

```xml
                            android:textColor="@android:color/black"/>
                </androidx.cardview.widget.CardView>
        </RelativeLayout>


        <RelativeLayout
            android:layout_height="wrap_content"
            android:layout_width="wrap_content"
            android:layout_gravity="start|bottom">
            <androidx.cardview.widget.CardView
                android:layout_height="wrap_content"
                android:layout_width="120dp"
                android:id="@+id/update_card"
                android:visibility="invisible"
                app:cardElevation="10dp"
                app:cardCornerRadius="5dp"
                app:cardBackgroundColor="@color/colorPrimaryDark"
                android:layout_margin="20dp">
                <TextView
                    android:layout_height="wrap_content"
                    android:layout_width="wrap_content"
                    android:layout_margin="10dp"
                    android:layout_gravity="center"
                    android:text="Previous"
                    android:textColor="@android:color/black"/>
            </androidx.cardview.widget.CardView>
        </RelativeLayout>


        <RelativeLayout
            android:layout_height="wrap_content"
            android:layout_width="wrap_content"
            android:layout_gravity="end">
            <androidx.cardview.widget.CardView
                android:layout_height="wrap_content"
                android:layout_width="wrap_content"
                android:id="@+id/fab2"
                android:visibility="visible"
                app:cardElevation="10dp"
                app:cardCornerRadius="5dp"
                app:cardBackgroundColor="@color/colorPrimaryDark"
                android:layout_margin="20dp">
                <TextView
                    android:layout_height="wrap_content"
                    android:layout_width="wrap_content"
                    android:layout_margin="10dp"
                    android:layout_gravity="center"
                    android:text="Save and upload"
                    android:textColor="@android:color/black"/>
            </androidx.cardview.widget.CardView>
        </RelativeLayout>

</androidx.coordinatorlayout.widget.CoordinatorLayout>




<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    android:layout_height="match_parent"
```

```xml
    android:layout_width="match_parent"
xmlns:android="http://schemas.android.com/apk/res/android">

<RelativeLayout
    android:layout_width="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:theme="@style/Theme.AppCompat.Light.NoActionBar"
    android:layout_height="match_parent">

    <!-- Add Toolbar -->
    <androidx.appcompat.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        android:background="?attr/colorPrimary"
        app:popupTheme="@style/ThemeOverlay.AppCompat.Light" />

    <androidx.appcompat.widget.AppCompatTextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/Sign_in_login"
        android:id="@+id/textview1"
        android:fontFamily="@font/lato_black"
        android:textColor="#366ce0"
        android:textSize="25sp"
        android:layout_marginStart="40dp"
        android:gravity="center"
        android:layout_marginTop="180dp">
    </androidx.appcompat.widget.AppCompatTextView>

    <View
        android:id="@+id/View1"
        android:layout_width="80dp"
        android:layout_height="7dp"
        android:layout_gravity="center"
        android:layout_marginTop="10dp"
        android:background="#366ce0"
        android:elevation="10dp"
        android:layout_alignStart="@id/textview1"
        android:layout_below="@+id/textview1">
    </View>

    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/card2"
        android:visibility="visible"
        android:layout_marginTop="30dp"
        android:layout_below="@id/View1"
        android:padding="20dp"
        app:cardUseCompatPadding="true"
        app:cardElevation="7dp"
        android:background="@drawable/circular_card"
        app:cardCornerRadius="7dp">

        <androidx.appcompat.widget.LinearLayoutCompat
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">
```

```xml
<androidx.appcompat.widget.AppCompatTextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/e_mail_text_view_login"
    android:fontFamily="@font/lato_black"
    android:layout_marginStart="20dp"
    android:textSize="20sp"
    android:textStyle="bold">
</androidx.appcompat.widget.AppCompatTextView>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="10dp"
        android:fontFamily="@font/lato_bold"
        android:maxLines="1"
        android:singleLine="true"
        android:layout_marginStart="20dp"
        android:layout_marginEnd="20dp"
        android:hint="@string/hint_email"
        android:inputType="textEmailAddress"
        android:textColor="@android:color/black"
        android:textColorHint="@android:color/white" />
</com.google.android.material.textfield.TextInputLayout>

<androidx.appcompat.widget.AppCompatTextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/e_mail_text_view_pass"
    android:layout_marginStart="20dp"
    android:fontFamily="@font/lato_black"
    android:textSize="20sp"
    android:textStyle="bold">
</androidx.appcompat.widget.AppCompatTextView>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/password"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="10dp"
        android:hint="@string/hint_password"
        android:fontFamily="@font/lato_bold"
        android:layout_marginStart="20dp"
        android:layout_marginEnd="20dp"
        android:inputType="textPassword"
        android:textColor="@android:color/black"
        android:textColorHint="@android:color/black" />
</com.google.android.material.textfield.TextInputLayout>
```

```xml
            <androidx.appcompat.widget.AppCompatCheckBox
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/radio1"
                android:text="@string/remember_me"
                android:fontFamily="@font/lato_black"
                android:layout_marginStart="20dp"
                android:textStyle="normal"
                android:textSize="15sp"
                android:textColor="#366ce0">
            </androidx.appcompat.widget.AppCompatCheckBox>

            <Button
                android:id="@+id/btn_reset_password"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginTop="0dp"
                android:layout_marginEnd="10dp"
                android:fontFamily="@font/lato_black"
                android:layout_gravity="end"
                android:background="@null"
                android:text="@string/btn_forgot_password"
                android:textAllCaps="false"
                android:textColor="@color/colorAccent" />

            <Button
                android:id="@+id/btn_login"
                android:layout_width="150dp"
                android:layout_gravity="center"
                android:layout_height="wrap_content"
                android:layout_marginTop="5dp"
                android:fontFamily="@font/lato_black"
                android:background="@drawable/button_circular"
                android:text="@string/btn_login"
                android:gravity="center"
                android:elevation="2dp"
                android:textColor="#ffff" />

            <Button
                android:id="@+id/btn_signup"
                android:layout_width="fill_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="20dip"
                android:background="@drawable/circular_coloured_button"
                android:text="@string/btn_link_to_register"
                android:fontFamily="@font/lato_black"
                android:textAllCaps="false"
                android:textColor="@color/fui_linkColor"
                android:textSize="15sp" />

        </androidx.appcompat.widget.LinearLayoutCompat>

    </androidx.cardview.widget.CardView>

    <!-- Add ProgressBar -->
    <ProgressBar
        android:id="@+id/loader1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```xml
                android:layout_centerInParent="true"
                android:visibility="gone" />

    </RelativeLayout>
</ScrollView>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true">

    <include
        layout="@layout/app_bar_main"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <com.google.android.material.navigation.NavigationView
        android:id="@+id/nav_view"
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        android:fitsSystemWindows="true"
        app:headerLayout="@layout/nav_header_main"
        app:menu="@menu/activity_main_drawer" />

</androidx.drawerlayout.widget.DrawerLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:theme="@style/Theme.AppCompat.Light.NoActionBar"
    android:layout_gravity="center"
    android:background="#ffff"
    android:fitsSystemWindows="true">

    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:gravity="center"
        android:orientation="vertical"
        android:padding="@dimen/activity_horizontal_margin">

        <ImageView
            android:contentDescription="USER_IMAGE"
```

```xml
            android:layout_width="@dimen/logo_w_h"
            android:layout_height="@dimen/logo_w_h"
            android:layout_gravity="center_horizontal"
            android:layout_marginBottom="10dp"
            android:src="@drawable/forget_pass" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:fontFamily="@font/lato_bolditalic"
            android:padding="10dp"
            android:text="@string/lbl_forgot_password"
            android:textColor="@android:color/holo_blue_light"
            android:textSize="20sp" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="10dp"
            android:gravity="center_horizontal"
            android:fontFamily="@font/lato_black"
            android:padding="@dimen/activity_horizontal_margin"
            android:text="@string/forgot_password_msg"
            android:textColor="@android:color/black"
            android:textSize="14sp" />

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <EditText
                android:id="@+id/email"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginBottom="10dp"
                android:layout_marginTop="20dp"
                android:focusable="true"
                android:fontFamily="@font/lato_light"
                android:hint="@string/hint_email"
                android:inputType="textEmailAddress"
                android:textColor="@android:color/black"
                android:textColorHint="@android:color/white" />
        </com.google.android.material.textfield.TextInputLayout>

        <!-- Login Button -->

        <Button
            android:id="@+id/btn_reset_password"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dip"
            android:background="@drawable/button_circular"
            android:fontFamily="@font/lato_black"
            android:text="@string/btn_reset_password"
            android:textColor="#ffff" />

        <Button
            android:id="@+id/btn_back"
```

```xml
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"
            android:background="@drawable/button_circular"
            android:fontFamily="@font/lato_black"
            android:text="@string/btn_back"
            android:textColor="#ffff" />

        <ProgressBar
            android:id="@+id/progressBar"
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:layout_gravity="center|bottom"
            android:layout_marginBottom="20dp"
            android:visibility="gone">
        </ProgressBar>

    </LinearLayout>

</RelativeLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent">

    <RelativeLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:theme="@style/Theme.AppCompat.Light.NoActionBar"
        android:orientation="vertical"
        android:background="#ededed"
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <androidx.appcompat.widget.AppCompatTextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/Sign_in_signup"
            android:id="@+id/textview3"
            android:fontFamily="@font/lato_black"
            android:textColor="#366ce0"
            android:textSize="25sp"
            android:layout_marginStart="40dp"
            android:gravity="center"
            android:layout_marginTop="180dp">
        </androidx.appcompat.widget.AppCompatTextView>

        <View
            android:id="@+id/View2"
            android:layout_width="80dp"
            android:layout_height="7dp"
            android:layout_gravity="center"
```

```xml
            android:layout_marginTop="10dp"
            android:background="#366ce0"
            android:elevation="10dp"
            android:layout_alignStart="@id/textview3"
            android:layout_below="@+id/textview3">
    </View>

    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/card3"
        android:visibility="visible"
        android:layout_marginTop="30dp"
        android:layout_below="@id/View2"
        android:padding="20dp"
        app:cardUseCompatPadding="true"
        app:cardElevation="7dp"
        android:background="@drawable/circular_card"
        app:cardCornerRadius="7dp">

        <androidx.appcompat.widget.LinearLayoutCompat
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <androidx.appcompat.widget.AppCompatTextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="@string/e_mail_text_view_login"
                android:layout_marginStart="20dp"
                android:fontFamily="@font/lato_black"
                android:textSize="20sp"
                android:textStyle="bold">
            </androidx.appcompat.widget.AppCompatTextView>

            <com.google.android.material.textfield.TextInputLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content">

                <EditText
                    android:id="@+id/email"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:hint="@string/email"
                    android:inputType="textEmailAddress"
                    android:fontFamily="@font/lato_bold"
                    android:maxLines="1"
                    android:layout_marginStart="20dp"
                    android:layout_marginEnd="20dp"
                    android:singleLine="true"
                    android:textColor="@android:color/black" />

            </com.google.android.material.textfield.TextInputLayout>

            <androidx.appcompat.widget.AppCompatTextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="@string/e_mail_text_view_pass"
                android:fontFamily="@font/lato_black"
```

```xml
                android:layout_marginStart="20dp"
                android:textSize="20sp"
                android:textStyle="bold">
        </androidx.appcompat.widget.AppCompatTextView>

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <EditText
                android:id="@+id/password"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:fontFamily="@font/lato_bold"
                android:focusableInTouchMode="true"
                android:hint="@string/hint_password"
                android:inputType="textPassword"
                android:layout_marginStart="20dp"
                android:layout_marginEnd="20dp"
                android:maxLines="1"
                android:singleLine="true"
                android:textColor="@android:color/black" />

        </com.google.android.material.textfield.TextInputLayout>

        <Button
            android:id="@+id/sign_up_button"
            style="?android:textAppearanceSmall"
            android:layout_width="150dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:gravity="center"
            android:fontFamily="@font/lato_black"
            android:layout_marginTop="15dp"
            android:background="@drawable/button_circular"
            android:text="@string/action_sign_in_btn"
            android:textColor="@android:color/black"
            android:textStyle="bold" />

        <Button
            android:id="@+id/btn_reset_password"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dip"
            android:fontFamily="@font/lato_black"
            android:background="@null"
            android:text="@string/btn_forgot_password"
            android:textAllCaps="false"
            android:textColor="#f2fe71"
            android:visibility="gone"/>

        <Button
            android:id="@+id/sign_in_button"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dp"
            android:fontFamily="@font/lato_black"
            android:layout_gravity="center"
            android:background="@drawable/circular_coloured_button"
```

```xml
                    android:text="@string/btn_link_to_login"
                    android:textAllCaps="false"
                    android:padding="2dp"
                    android:textColor="@color/white"
                    android:textSize="12sp" />

            </androidx.appcompat.widget.LinearLayoutCompat>

        </androidx.cardview.widget.CardView>

        <ProgressBar
            android:id="@+id/loader1"
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:layout_gravity="center|bottom"
            android:layout_marginBottom="20dp"
            android:visibility="gone">
        </ProgressBar>

    </RelativeLayout>

</ScrollView>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/Theme.MyApplication.AppBarOverlay">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?android:attr/actionBarSize"
            android:background="?android:attr/colorPrimary"
            app:popupTheme="@style/Theme.MyApplication.PopupOverlay" />
    </com.google.android.material.appbar.AppBarLayout>

    <ProgressBar
        android:id="@+id/loader1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:visibility="gone" />

    <ListView
        android:id="@+id/test_listview"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@id/appbar"
        android:layout_marginTop="16dp"
```

```xml
            android:layout_marginBottom="16dp"
            android:layout_marginStart="16dp"
            android:layout_marginEnd="16dp"
            android:divider="@android:color/darker_gray"
            android:dividerHeight="1dp" />

    <androidx.cardview.widget.CardView
        android:id="@+id/download_card"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/test_listview"
        android:layout_marginTop="16dp"
        android:layout_marginBottom="16dp"
        android:layout_marginStart="16dp"
        android:layout_marginEnd="16dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="4dp">


    </androidx.cardview.widget.CardView>

</RelativeLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".View.MainActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/Theme.MyApplication.AppBarOverlay">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?android:attr/actionBarSize"
            android:background="?android:attr/colorPrimary"
            app:popupTheme="@style/Theme.MyApplication.PopupOverlay">
            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="end"
                android:layout_marginEnd="15dp">

                <ImageButton
                    android:id="@+id/userImage2"
                    android:contentDescription="App rules and Usages!"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
```

```xml
                android:src="@drawable/ic_info_black_24dp"
                android:background="@color/colorPrimary"
                android:tooltipText="App rules and Usages!"
                android:paddingEnd="10dp"
                android:scaleType="fitCenter"
                android:layout_gravity="center_horizontal">

            </ImageButton>

        </LinearLayout>
    </androidx.appcompat.widget.Toolbar>

</com.google.android.material.appbar.AppBarLayout>

<include layout="@layout/content_main" />


</FrameLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:background="@android:color/white"
    android:layout_gravity="center"
    android:orientation="vertical">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/Theme.MyApplication.AppBarOverlay"
        android:id="@+id/tooltst">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbartst"
            app:navigationIcon="@drawable/ic_keyboard_arrow_left_black_24dp"
            android:layout_width="match_parent"
            android:layout_height="?android:attr/actionBarSize"
            android:background="?android:attr/colorPrimary"
            app:popupTheme="@style/Theme.MyApplication.PopupOverlay">
        </androidx.appcompat.widget.Toolbar>

    </com.google.android.material.appbar.AppBarLayout>

    <TableLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <TableRow
            android:id="@+id/tableRow1"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dp"
```

```xml
            android:gravity="center" >

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:gravity="center"
                android:text="Student Details"
                android:fontFamily="@font/lato_black"
                android:textColor="@color/bg_main"
                android:textSize="14sp"
                android:textStyle="bold" />
        </TableRow>
</TableLayout>

<TableLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <TableRow
        android:layout_height="match_parent"
        android:layout_width="wrap_content"
        android:paddingBottom="15dp"
        android:paddingTop="15dp"
        android:gravity="center">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/USERIMAGE"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:src="@drawable/user">

        </de.hdodenhof.circleimageview.CircleImageView>
    </TableRow>
</TableLayout>


    <TableRow
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingStart="20dp"
        android:paddingEnd="20dp">

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/bg_main"
            android:text="StudentID"
            android:fontFamily="@font/lato_black"
            android:padding="5dp"
            android:textSize="14sp"
            android:textColor="@android:color/white"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/StudentID"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
```

```xml
            android:layout_weight="1"
            android:background="@color/colorAccent"
            android:fontFamily="@font/lato_bold"
            android:text="Details not added yet"
            android:padding="5dp"
            android:textSize="14sp"
            android:textColor="@android:color/white"
            android:textStyle="bold" />
    </TableRow>

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingStart="20dp"
        android:paddingEnd="20dp">

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/bg_main"
            android:fontFamily="@font/lato_black"
            android:text="Semester"
            android:padding="5dp"
            android:textSize="14sp"
            android:textColor="@android:color/white"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/SemID"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/colorAccent"
            android:fontFamily="@font/lato_bold"
            android:text="Details not added yet"
            android:padding="5dp"
            android:textSize="14sp"
            android:textColor="@android:color/white"
            android:textStyle="bold" />
    </TableRow>

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingStart="20dp"
        android:paddingEnd="20dp">

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:fontFamily="@font/lato_black"
            android:background="@color/bg_main"
            android:text="Branch"
            android:padding="5dp"
            android:textSize="14sp"
            android:textColor="@android:color/white"
            android:textStyle="bold" />
```

```xml
            <TextView
                android:id="@+id/Branchid"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:background="@color/colorAccent"
                android:fontFamily="@font/lato_bold"
                android:text="Details not added yet"
                android:padding="5dp"
                android:textSize="14sp"
                android:textColor="@android:color/white"
                android:textStyle="bold" />
        </TableRow>

        <TableRow
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:paddingStart="20dp"
            android:paddingEnd="20dp">

            <TextView
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:background="@color/bg_main"
                android:fontFamily="@font/lato_black"
                android:text="Section"
                android:padding="5dp"
                android:textSize="14sp"
                android:textColor="@android:color/white"
                android:textStyle="bold" />

            <TextView
                android:id="@+id/SectionID"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:background="@color/colorAccent"
                android:fontFamily="@font/lato_bold"
                android:text="Details not added yet"
                android:padding="5dp"
                android:textSize="14sp"
                android:textColor="@android:color/white"
                android:textStyle="bold" />
        </TableRow>

    <TableLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >

        <TableRow
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dp"
            android:gravity="center" >

            <TextView
                android:id="@+id/this_Section"
```

```xml
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:gravity="center"
                    android:text="This test marks is: "
                    android:fontFamily="@font/lato_black"
                    android:textColor="@color/bg_main"
                    android:textSize="14sp"
                    android:textStyle="bold" />
        </TableRow>

    </TableLayout>

</LinearLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin">


    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/scrollView"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:fillViewport="true">

    <LinearLayout
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:weightSum="1"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true">


        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="20dp"
            android:orientation="horizontal">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/textView3"
                android:fontFamily="@font/lato_black"
```

```xml
        android:text="QUESTION"
        android:textSize="30sp"
        android:layout_marginEnd="20dip"/>

    <TextView
        android:layout_width="156dp"
        android:layout_height="wrap_content"
        android:fontFamily="@font/lato_black"
        android:id="@+id/questionNumber"
        android:layout_marginEnd="291dp"
        android:text="1"
        android:textSize="30sp"/>

</LinearLayout>
<EditText
    android:layout_width="match_parent"
    android:layout_height="150dp"
    android:id="@+id/questionView"
    android:fontFamily="@font/lato_blackitalic"
    android:layout_marginBottom="30sp"
    android:hint="Type your question here" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="30dp"
    android:weightSum="1">
    <RadioButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="A"
        android:id="@+id/aRadio" />

    <EditText
        android:inputType="text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="@font/lato_blackitalic"
        android:hint="OPTION A"
        android:id="@+id/aText"
        android:layout_marginStart="20sp" />

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="30sp"
    android:weightSum="1">


    <RadioButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="B"
        android:id="@+id/bRadio" />

    <EditText
        android:inputType="text"
        android:layout_width="match_parent"
```

```xml
                android:layout_height="wrap_content"
                android:hint="OPTION B"
                android:fontFamily="@font/lato_blackitalic"
                android:id="@+id/bText"
                android:layout_marginStart="20sp" />

        </LinearLayout>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="30sp"
            android:weightSum="1">
            <RadioButton
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="C"
                android:id="@+id/cRadio" />

            <EditText
                android:inputType="text"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:hint="OPTION C"
                android:fontFamily="@font/lato_blackitalic"
                android:id="@+id/cText"
                android:layout_marginStart="20sp" />

        </LinearLayout>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="30sp"
            android:weightSum="1">

            <RadioButton
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="D"
                android:id="@+id/dRadio" />

            <EditText
                android:inputType="text"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:hint="OPTION D"
                android:fontFamily="@font/lato_blackitalic"
                android:id="@+id/dText"
                android:layout_marginStart="20sp" />
        </LinearLayout>

    </LinearLayout>
</ScrollView>
</RelativeLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
```

```xml
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    android:layout_gravity="center"
    android:orientation="vertical"
    tools:context=".View.MainActivity"
    tools:showIn="@layout/app_bar_main"
    android:layout_marginTop="?android:attr/actionBarSize">


    <androidx.appcompat.widget.LinearLayoutCompat
        android:id="@+id/li1"
        android:layout_width="match_parent"
        android:layout_height="288dp"
        android:orientation="vertical">

        <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="230dp"
            android:layout_gravity="center_horizontal|center"
            android:background="@drawable/circular_card"
            android:backgroundTint="@android:color/holo_blue_light">


            <de.hdodenhof.circleimageview.CircleImageView
                android:id="@+id/card1"
                android:layout_width="180dp"
                android:layout_height="180dp"
                android:layout_gravity="center_horizontal"
                android:paddingTop="25dp"
                android:scaleType="centerCrop"
                android:src="@drawable/icon1">

            </de.hdodenhof.circleimageview.CircleImageView>

        </androidx.cardview.widget.CardView>

        <TextView
            android:id="@+id/text_user_card"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:fontFamily="@font/lato_black"
            android:padding="7dp"
            android:gravity="center"
            android:text="@string/Welcome"
            android:textAllCaps="true"
            android:textColor="@color/fui_bgGitHub"
            android:textSize="15sp" />


    </androidx.appcompat.widget.LinearLayoutCompat>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```xml
        android:orientation="vertical"
        android:layout_marginTop="50dp"
        app:layout_anchorGravity="bottom"
        android:layout_gravity="bottom"
        android:visibility="gone"
        app:layout_anchor="@id/li1">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="Recent Activities"
            android:textColor="@android:color/holo_red_light"
            android:fontFamily="@font/lato_black"
            android:textSize="20sp">

        </TextView>


        <androidx.recyclerview.widget.RecyclerView
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="10"
            android:orientation="vertical"
            android:id="@+id/recent_activity_list"
            android:divider="@null"
            android:dividerHeight="0dp">

        </androidx.recyclerview.widget.RecyclerView>
    </LinearLayout>

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="40dp"
        android:id="@+id/chatHead"
        android:layout_marginEnd="17dp"
        android:backgroundTint="#ffff"
        android:src="@drawable/chat"
        app:icon="@drawable/chat"
        android:background="@drawable/circular_coloured_button"
        android:tint="@android:color/black"
        android:scaleType="center"
        app:fabSize="normal"
        app:maxImageSize="56dp"
        android:elevation="10dp"
        android:layout_gravity="bottom|end">

    </com.google.android.material.floatingactionbutton.FloatingActionButton>


</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:fitsSystemWindows="true">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/Theme.MyApplication.AppBarOverlay"
        android:id="@+id/tooltst">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbartst"
            app:navigationIcon="@drawable/ic_keyboard_arrow_left_black_24dp"
            android:layout_width="match_parent"
            android:layout_height="?android:attr/actionBarSize"
            android:background="?android:attr/colorPrimary"
            app:popupTheme="@style/Theme.MyApplication.PopupOverlay">
        </androidx.appcompat.widget.Toolbar>

    </com.google.android.material.appbar.AppBarLayout>


    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fab"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|end"
        android:layout_margin="@dimen/fab_margin"
        android:background="#ff5252"
        android:visibility="visible"
        android:backgroundTint="#ff5252"
        android:src="@android:drawable/ic_menu_add" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginTop="?android:attr/actionBarSize">


    <ListView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/test_listview"
        android:divider="@android:color/transparent"
        android:dividerHeight="0dp">
```

```xml
    </ListView>

    <RelativeLayout
        android:layout_height="wrap_content"
        android:layout_width="wrap_content"
        android:layout_gravity="end|bottom">

        <androidx.cardview.widget.CardView
            android:layout_height="wrap_content"
            android:layout_width="wrap_content"
            android:id="@+id/download_card"
            android:visibility="invisible"
            app:cardElevation="10dp"
            app:cardCornerRadius="5dp"
            app:cardBackgroundColor="@color/colorPrimaryDark"
            android:layout_margin="20dp">

            <TextView
                android:layout_height="wrap_content"
                android:layout_width="wrap_content"
                android:layout_margin="10dp"
                android:text="Download"
                android:textColor="@android:color/white"/>
        </androidx.cardview.widget.CardView>
    </RelativeLayout>


</FrameLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/layout_root"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:padding="10dp" >

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="@font/lato_black"
        android:text="Save test "
        android:textAppearance="?android:attr/textAppearanceLarge" />

    <EditText
        android:id="@+id/editTextDialogUserInput"
        android:layout_width="match_parent"
        android:fontFamily="@font/lato_bold"
        android:layout_height="wrap_content" >
```

```xml
            <requestFocus />

    </EditText>

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="Test Name"
        android:fontFamily="@font/lato_black"
        android:text="Specify Time"
        android:textAppearance="?android:attr/textAppearanceLarge" />

    <EditText
        android:id="@+id/editTextDialogUserInput1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="@font/lato_bold"
        android:inputType="number"
        android:hint="Enter time in digit">

    </EditText>

</LinearLayout>


<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingTop="@dimen/activity_horizontal_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin">

    <TextView
        android:text="questions"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/questionTextView"
        android:fontFamily="@font/lato_black"
        android:textStyle="bold"
        android:textSize="28sp"
        android:textColor="#eb346e"
        android:paddingStart="10dp"
        android:layout_gravity="start"
        android:paddingBottom="10dp"/>

    <RadioGroup
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="start|center_vertical"
        android:layout_marginStart="20dp"
        android:scrollbarSize="20dp"
        android:gravity="center"
```

```xml
        android:id="@+id/radioGroup">

        <RadioButton
            android:text="Rb1"
            android:textSize="20sp"
            android:layout_width="match_parent"
            android:fontFamily="@font/lato_black"
            android:layout_height="wrap_content"
            android:id="@+id/radioButton"
            android:paddingBottom="10dp">
        </RadioButton>
        <RadioButton
            android:textSize="20sp"
            android:text="Rb2"
            android:fontFamily="@font/lato_black"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/radioButton2" />

        <RadioButton
            android:textSize="20sp"
            android:text="Rb3"
            android:layout_width="match_parent"
            android:fontFamily="@font/lato_black"
            android:layout_height="wrap_content"
            android:id="@+id/radioButton3" />
        <RadioButton
            android:textSize="20sp"
            android:text="Rb4"
            android:layout_width="match_parent"
            android:fontFamily="@font/lato_black"
            android:layout_height="wrap_content"
            android:id="@+id/radioButton4" />
        <RadioButton
            android:textSize="20sp"
            android:text="Clear Selection"
            android:layout_width="match_parent"
            android:fontFamily="@font/lato_black"
            android:layout_height="wrap_content"
            android:id="@+id/radioButton5" />

    </RadioGroup>
</LinearLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="@dimen/nav_header_height"
    android:background="@drawable/side_nav_bar"
    android:gravity="bottom"
    android:orientation="vertical"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
```

```xml
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin">

    <de.hdodenhof.circleimageview.CircleImageView
        android:id="@+id/imageView"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:contentDescription="@string/nav_header_desc"
        android:paddingTop="@dimen/nav_header_vertical_spacing"
        android:src="@drawable/user" />

    <TextView
        android:id="@+id/text_user_name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingTop="@dimen/nav_header_vertical_spacing"
        android:fontFamily="@font/lato_regular"
        android:text="UserName"
        android:textAppearance="@style/TextAppearance.AppCompat.Body1" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="@font/lato_bold"
        android:text="Email-Id" />

</LinearLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

    <androidx.cardview.widget.CardView
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:cardCornerRadius="8dp"
        app:cardElevation="6dp"
        app:cardPreventCornerOverlap="false"
        app:cardUseCompatPadding="true">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="88dp"
            android:orientation="horizontal"
            android:gravity="center"
            android:background="@color/colorPrimary">
            <ImageView
                android:layout_width="0dp"
                android:layout_weight="1"
                android:layout_height="40dp"
```

```xml
                android:src="@drawable/ic_appicon"
                android:id="@+id/item_imageView"
                android:gravity="center"/>
            <TextView
                android:layout_width="0dp"
                android:layout_weight="4"
                android:layout_height="wrap_content"
                android:text="Quiz1"
                style="@style/CardStyle"
                android:id="@+id/item_textView" />
            <Button
                android:layout_margin="5dp"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:gravity="center"
                android:fontFamily="@font/lato_black"
                android:id="@+id/item_button"
                android:background="@drawable/button_circular"
                android:layout_alignParentEnd="true"/>

        </LinearLayout>
    </androidx.cardview.widget.CardView>

</LinearLayout>
```